CONVERGENT ALGORITHMS IN SIMULATION OPTIMIZATION

A Thesis
Presented to
The Academic Faculty

by

Liujia Hu

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Industrial and Systems Engineering

Georgia Institute of Technology
May 2015

CONVERGENT ALGORITHMS IN SIMULATION OPTIMIZATION

Approved by:

Professor Sigrún Andradóttir, Advisor
School of Industrial and Systems Engineering
Georgia Institute of Technology

Professor Shabbir Ahmed
School of Industrial and Systems Engineering
Georgia Institute of Technology

Professor David M. Goldsman
School of Industrial and Systems Engineering
Georgia Institute of Technology

Professor Seong-Hee Kim
School of Industrial and Systems Engineering
Georgia Institute of Technology

Professor Richard M. Fujimoto
School of Computational Science and Engineering
Georgia Institute of Technology

Date Approved: 27 March 2015

To my parents,

Guangqin Liu and Xinming Hu.

ACKNOWLEDGEMENTS

I want to thank my advisor, Dr. Sigrún Andradóttir for guiding and supporting me over the past four years. Without her help, I would not have made so much progress and completed my thesis.

I would like to thank my thesis committee members, Dr. Shabbir Ahmed, Dr. David M. Goldsman, Dr. Seong-Hee Kim, and Dr. Richard M. Fujimoto, for their guidance and valuable comments throughout this whole process.

I would like to thank my friends, classmates, and colleagues for their support during my PhD study. We really had a great time in Atlanta over the past five years.

Finally, I would like to thank my parents for their unconditional love. Without their support, I would not have had the chance to pursue and obtain my doctoral degree.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

It is frequently the case that deterministic optimization models could be made more practical by explicitly incorporating uncertainty. The resulting stochastic optimization problems are in general more difficult to solve than their deterministic counterparts, because the objective function cannot be evaluated exactly and/or because there is no explicit relation between the objective function and the corresponding decision variables. This thesis develops random search algorithms for solving optimization problems with continuous decision variables when the objective function values can be estimated with some noise via simulation. Our algorithms will maintain a set of sampled solutions, and use simulation results at these solutions to guide the search for better solutions.

In the first part of the thesis, we propose an Adaptive Search with Resampling and Discarding (ASRD) approach for solving continuous stochastic optimization problems. Our ASRD approach is a framework for designing provably convergent algorithms that are adaptive both in seeking new solutions and in keeping or discarding already sampled solutions. The framework is an improvement over the Adaptive Search with Resampling (ASR) method of Andradóttir and Prudius in that it spends less effort on inferior solutions (the ASR method does not discard already sampled solutions). We present conditions under which the ASRD method is convergent almost surely and carry out numerical studies aimed at comparing the algorithms. Moreover, we show that whether it is beneficial to resample or not depends on the problem, and analyze when resampling is desirable. Our numerical results show that the ASRD approach makes substantial improvements on ASR, especially for difficult problems with large numbers of local optima.

In traditional simulation optimization problems, noise is only involved in the objective functions. However, many real world problems involve stochastic constraints. Such problems are more difficult to solve because of the added uncertainty about feasibility. The second part of the thesis presents an Adaptive Search with Discarding and Penalization (ASDP) method for solving continuous simulation optimization problems involving stochastic constraints. Rather than addressing feasibility separately, ASDP utilizes the penalty function method from deterministic optimization to convert the original problem into a series of simulation optimization problems without stochastic constraints. We present conditions under which the ASDP algorithm converges almost surely from inside the feasible region, and under which it converges to the optimal solution but without feasibility guarantee. We also conduct numerical studies aimed at assessing the efficiency and tradeoff under the two different convergence modes.

Finally, in the third part of the thesis, we propose a random search method named Gaussian Search with Resampling and Discarding (GSRD) for solving simulation optimization problems with continuous decision spaces. The method combines the ASRD framework with a sampling distribution based on a Gaussian process that not only utilizes the current best estimate of the optimal solution but also learns from past sampled solutions and their objective function observations. We prove that our GSRD algorithm converges almost surely, and carry out numerical studies aimed at studying the effects of utilizing the Gaussian sampling strategy. Our numerical results show that the GSRD framework performs well when the underlying objective function is multi-modal. However, it takes much longer to sample solutions, especially in higher dimensions.

CHAPTER I

INTRODUCTION

Traditional optimization helps us make better decisions when all the information and data are available and deterministic, whereas simulation assists us to better understand possible outcomes when we do not have all the information and when uncertainty is involved. Simulation optimization algorithms are designed to find an optimal or near-optimal solution when the objective function needs to be evaluated through simulation.

This thesis develops provably convergent simulation optimization algorithms for solving optimization problems involving continuous decision variables, constraints, and uncertainties. Such problems are of interest, mainly, because many real-world optimization problems are too complicated to compute exact objective function as well as constraint values. However, they are generally hard to solve due to the uncertainties involved in the problem formulations, and often possess little structure that could be utilized by traditional optimization techniques.

Specifically, we are concerned with solving the following two problems:

(I) Simulation Optimization Problem:

$$\sup_{\theta \in \Theta} f(\theta) = E[h(\theta, X(\omega))]; \tag{1}$$

(II) Simulation Optimization Problem with Stochastic Constraints:

$$\begin{aligned} \sup_{\theta \in \Theta} \quad & f(\theta) = E[h(\theta, X(\omega))] \\ \text{subject to} \quad & g_j(\theta) = E[u_j(\theta, Y_j(\omega))] \leq b_j, \ j \in \mathcal{C}. \end{aligned} \tag{2}$$

We assume the feasible region Θ is some set in an s-dimensional space $\mathbb{R}^s$, $\mathcal{C}$ is a finite set of indexes, E denotes the mathematical expectation operation, $X(\omega)$

and $Y_j(\omega)$, $j \in \mathcal{C}$, are random elements defined on some probability space $(\Omega, \Sigma, \mathbb{P})$, and h, u_j, $j \in \mathcal{C}$, are deterministic, real-valued functions and measurable in the second argument. We allow the feasible region Θ to be uncountable and assume $f(\theta)$ and $g_j(\theta)$ at any $\theta \in \Theta$ and $j \in \mathcal{C}$ cannot be evaluated exactly. Thus, if there are deterministic constraints, we assume they are incorporated in the feasible region Θ. The reason for not bringing the deterministic constraints out is mainly that no additional observations need to be obtained to determine feasibility, whereas multiple observations are required to reduce the noise involved in stochastic constraints. Let f^* be the optimal objective value of both optimization problem (1) and (2). Finally, let $\bar{f} = \sup_{\theta \in \Theta} f(\theta)$ in optimization problem (2). Assume $f^* < \infty$ in both problems, and $\bar{f} < \infty$ in problem (2).

Andradóttir and Prudius [13] proposed an efficient Adaptive Search with Resampling (ASR) approach to solve the simulation optimization problem (1) when the feasible region Θ is continuous. The method adaptively samples new solutions in Θ, decides whether or not to accept a newly sampled solution, and resamples previously sampled and accepted solutions. The main advantages of this method are: 1) It is guaranteed to converge almost surely. 2) It is adaptive, so the sampling strategy can be based on all the information collected by the method so far. 3) It has an acceptance criterion (to avoid spending excessive effort on inferior solutions) and exhibits good empirical performance. However, the set of accepted sampled points keeps growing in the ARS algorithm because there is no scheme to discard inferior points that are previously accepted. With time, the computational effort of resampling inferior points could be huge.

In the first part of this thesis, we develop an Adaptive Search with Resampling and Discarding (ASRD) method that efficiently discards some of the originally accepted sampled points in the ARS algorithm of Andradóttir and Prudius [13]. In our new algorithm, almost sure convergence is guaranteed, the sampling strategy remains

adaptive, efficient acceptance criteria can be incorporated to avoid spending excessive effort on inferior solutions, and the discarding scheme saves simulation budgets, as well as memory space, spent on inferior points. Moreover, we discover that whether it is beneficial to resample or not depends on the underlying problem, and we derive an indicator to assist us in choosing whether to resample or not.

When stochastic constraints are involved in simulation optimization problems, one natural way to solve (2) is to apply a framework designed to solve (1) and use estimated constraint function values to test feasibility, since the only difference between (1) and (2) is the stochastic constraints. However, in the second part of the thesis, we show that additional efforts should be made besides purely testing feasibility based on sample average approximation. We also propose a provably convergent algorithm called Adaptive Search with Discarding and Penalization (ASDP) to solve (2). We use a sequence of positive real numbers to dynamically penalize sampled points that appear to be infeasible or whose feasibility is ambiguous (meaning that they appear to be either feasible or infeasible but very close to the boundary of the feasible region). We also use two other sequences of non-negative real numbers to discard points that are likely to be infeasible and/or inferior. The unique feature of our ASDP algorithm is that as the number of iterations goes to infinity, the algorithm converges almost surely from *inside the feasible region*. No existing algorithms have this property as far as we know.

Gaussian processes have been widely used in stochastic modeling, optimization, machine learning, simulation, statistics, etc. Sun, Hong, and Hu [65] proposed a fast fitted Gaussian process constructed based on previously evaluated solutions, and designed a sampling distribution based on the Gaussian process that can automatically balance the tradeoff between exploitation and exploration in discrete decision spaces. In the third part of this thesis, we extend the sampling approach for discrete decision spaces of Sun, Hong, and Hu [65], combine it with the ASRD optimization framework,

and develop a new random search algorithm called Gaussian Search with Resampling and Discarding (GSRD) that is aimed at solving simulation optimization problems on continuous decision spaces. We prove that GSRD converges almost surely to the global optimal solution and carry out numerical analysis to study the pros and cons of Gaussian sampling relative to point-based adaptive search.

The remainder of this thesis is organized as follows. In Chapter 2 we briefly review the existing literature on simulation optimization. In Chapter 3 we present the ASRD framework, prove that ASRD converges with probability one under some mild assumptions, and conduct a numerical study aimed at assessing the effects of the discarding procedure. In Chapter 4, we develop the ASDP method for solving constrained simulation optimization problems, prove its almost sure convergence both from inside the feasible region and without feasibility guarantee, and test the ASDP algorithm on various problems and obtain promising results. In Chapter 5, we propose the GSRD approach, prove its almost sure convergence, and analyze the effects of Gaussian sampling. Finally, in Chapter 6 we summarize our main contributions and provide some future research directions.

CHAPTER II

LITERATURE REVIEW

The simulation optimization problem is concerned with finding optimal decision parameters for a system that involves uncertainty, where optimality is measured by either maximizing or minimizing the expectation of the objective function. One fundamental assumption in simulation optimization is that the objective function is not available directly, but needs to be estimated via simulation. This field has been extensively studied in the past three decades. The main approaches in the simulation literature to solve this type of problems are ranking and selection, response surface methodology, random search, stochastic approximation, sample average approximation, and model-based methods.

There are a number of surveys on this topic with focus on different approaches. Goldsman and Nelson [24] and Swisher, Jacobson, and Yucesan [67] provide a comprehensive review on ranking and selection and multiple comparisons. Andradóttir [7] provides a review on gradient estimation and stochastic approximation. Also, a comprehensive review on random search methods and the corresponding convergence results can be found in Andradóttir [10]. Carson and Maria [17], Fu [21], Swisher et al. [66], Fu, Glover, and April [22], and Hong and Nelson [33] provide general reviews of the main approaches used for simulation optimization, including both algorithms and convergence results, and discuss implementation in commercial software and future research directions.

Most existing simulation optimization methods are designed for solving (1) or (2) when the feasible region is either discrete or continuous. In Section 2.1, we discuss approaches designed for solving discrete simulation optimization problems,

with emphasis on random search. In Section 2.2, we briefly review methods for solving continuous simulation optimization problems.

2.1 Discrete Feasible Region

In recent years, many simulation optimization algorithms aiming at solving problem (1) when Θ is discrete have been developed. One popular approach to solve discrete simulation optimization problems is to use random search. Random search (RS) is a family of methods do not require the gradient of the underlying objective function to be optimized. It can be used on functions that have little known structure. RS only requires objective function estimates to be available.

Kirkpatrick, Gelatt, and Vecchi [46] utilized the idea of simulated annealing (SA) to solve (deterministic) combinatorial optimization problems. Since then, the use of SA in solving discrete simulation optimization problems has been studied extensively. Gelfand and Mitter [23] presented a convergence analysis of an SA algorithm designed to solve (1). Their modification of SA is provably convergent when the noise in the objective function estimates is normally distributed with mean zero and variance decreasing asymptotically faster than the cooling schedule. Similarly, Fox and Heine [19] showed that the SA algorithm with noisy objective function estimates converges in probability under weak conditions, and Gutjahr and Pflug [27] also obtained convergence results for the SA algorithm when the function observations are disturbed by random noise. Alrefaei and Andradóttir [1] proposed two variants of the SA algorithm to solve the optimization problem (1) that use a constant temperature rather than a decreasing temperature. The two modifications are provably convergent and appear to be more efficient numerically than the SA methods in [19, 23, 27].

Yan and Mukai [72] proposed the stochastic ruler method. The method uses a predetermined uniform random variable ("stochastic ruler") to generate a sequence of solution estimates that forms a non-stationary, strongly ergodic Markov chain

under mild conditions, and the Markov chain converges to the optimal solution in probability. Alrefaei and Andradóttir [2, 3] proposed modifications of the stochastic ruler method. The basic idea is to use either the number of visits to every state by the induced Markov chain or the best average estimated objective function value obtained from all the previous observations of the objective function values as the estimate of the optimal solution. These algorithms are almost surely convergent to the optimal solution set and numerically more efficient than the original algorithm.

Gong, Ho, and Zhai [25, 26] proposed the stochastic comparison method. Similar to the stochastic ruler method, their method solves an alternative optimization problem instead of the original optimization problem. In each iteration, they compare the observations of the objective function at two different parameter values (rather than using a stochastic ruler like Yan and Mukai [72]). Gong, Ho, and Zhai [26] showed that the original and alternative optimization problems they consider are equivalent under some conditions, and that their method converges in probability to a global solution of the alternative optimization problem. Andradóttir [4, 6] also developed stochastic comparison methods where the comparison is carried out with an estimate of the objective function value at the current solution, rather than a stochastic ruler. The methods in [4, 6] both converge almost surely to the global optima. Andradóttir [8] presented an almost surely convergent variant of the stochastic comparison method of Gong, Ho, and Zhai [25], and discussed its convergence rate.

Shi and Ólafsson [63, 64] proposed the Nested Partitions (NP) method to solve discrete optimization problems, where [63] discussed deterministic optimization problems and [64] addressed stochastic optimization problems. The method partitions the feasible region into subregions, adaptively identifies the most promising region, further partitions the current most promising region and merges the other subregions at each iteration, and backtracks if a better solution is found outside the most promising region. Pichitlamken and Nelson [56] adopted the spirit of the NP method of Shi and

Ólafsson [64], and proposed a combined procedure that consists of a global guidance system, a selection-of-the-best procedure, and local improvement for optimization via simulation.

Hong and Nelson [34] proposed a locally convergent algorithm called COMPASS for discrete-event simulation optimization problems with integer ordered decision variables. It is a random search method that adaptively focuses its sampling on the current most promising region, which is defined as all the feasible points that are closer to the sampled point with the highest estimated objective function value than to other points sampled so far.

More recently, Xu, Nelson, and Hong [69] developed an adaptive hyperbox algorithm (AHA) for solving discrete simulation optimization problems and prove that it is a locally convergent algorithm. Compared to COMPASS, the AHA algorithm constructs a hyperbox as the most promising region at each iteration, and it is more efficient in high-dimensional problems. Xu [68] proposed a random search method using stochastic kriging combined with AHA to solve discrete simulation optimization problems. Sun, Hong, and Hu [65] combined the ideas of kriging meta-modeling (Jones [43]) and BEESE (Andradóttir and Prudius [12]) to derive a sampling distribution, that automatically balances the tradeoff between exploitation and exploration. The sampling distribution involves a Gaussian process that is based on previously sampled solutions, and Sun, Hong, and Hu [65] incorporate this sampling distribution into a random search algorithm.

When Θ is countably infinite, Andradóttir [9] discussed how the estimate of the optimal solution should be chosen to guarantee the almost sure convergence of a class of simulation optimization algorithms, and Hong and Nelson [36] developed a random search framework to find local optima.

When Θ is discrete and stochastic constraints are involved (refer to (2)), ranking and selection can be used to provide a guaranteed probability of selecting the best

system (see Andradóttir and Kim [11]). Similarly, the Optimal Computing Budget Allocation (OCBA) approach can be used to maximize the probability of selecting the best system. Some works in this area include Kabirian and Ólafsson [44], Andradóttir and Kim [11], Lee et al. [49], Healey, Andradóttir, and Kim [31], Hunter and Pasupathy [42], etc. On the other hand, Li, Sava, and Xie [50] proposed a random search method where stochastic constraints are taken into account in an augmented performance function via an increasing penalty factor, and Park and Kim [55] proposed a penalty function with memory (PFM) method that successfully handles the simulation optimization problem with stochastic constraints (2).

2.2 Continuous Feasible Region

The sample average approximation (SAA) approach is a popular class of methods that has been developed to solve the stochastic optimization problem (1) when Θ is continuous (SAA can also be used when Θ is discrete). The main idea of SAA is to generate a random sample of objective function observations and approximate the expected value function by the corresponding sample average function. Some representative works include Healy and Schruben [32], Robinson [60], Shapiro and Wardi [62], and Kleywegt, Shapiro, and Homem-de-Mello [47].

In addition, the application of stochastic approximation methods to solve the simulation optimization problem (1) has been studied extensively. The work includes articles by Robbins and Monro [59], Kiefer and Wolfowitz [45], Andradóttir [5], Kushner and Yin [48], Polyak and Juditsky [57], and Nemirovski et al. [51]. An important step in applying both SAA and stochastic approximation methods is to estimate the gradient of the objective function. A thorough review of works focused on gradient estimation can be found in Fu [20].

A few random search algorithms have been proposed for solving continuous simulation optimization problems (i.e., Θ is uncountable). Yakowitz and Lugosi [71] developed a global random search method with resampling, and Yakowitz [70] proposed a random search method incorporating stochastic approximation. Norkin, Pflug, and Ruszczyński [53] presented and analyzed a stochastic branch-and-bound method. Baumert and Smith [15] proposed a method based on pure random search that estimates the objective function value at each solution by averaging all observations within a certain distance from that solution, and ensured that the method converges in probability by shrinking this distance gradually. Andradóttir and Prudius [13] proposed the adaptive search with resampling (ASR) approach, and also studied the deterministic and stochastic shrinking ball methods that resemble the shrinking ball method of Baumert and Smith [15].

Model-based methods form a recently developed class of randomized search techniques. Most algorithms in this class involve generating candidate solutions according to a specific probability measure on the solution space, then updating the probability measure based on the candidate solutions generated in the previous step and their estimated performance. The purpose is to fully utilize the available information throughout the simulation process to find the optimal solution set. Specifically, Rubinstein and Kroese [61] proposed the cross entropy (CE) approach, and Hu, Fu, and Marcus [38] developed the stochastic model reference adaptive search (SMRAS).

When Θ is discrete or continuous with stochastic constraints, refer to problem (2), the sample average approximation (SAA) approach also can be used to handle the stochastic constraints. For example, Pagnoncelli, Ahmed, and Shapiro [54] studied how to use SAA to obtain good candidate solutions for chance constrained problems and discussed the convergence properties of the resulting problems, and Dentcheva and Ruszczynski [18] introduced benchmark stochastic optimization problems involving stochastic dominance constraints, and developed optimality conditions, duality

theory, etc.

CHAPTER III

ADAPTIVE SEARCH WITH DISCARDING FOR CONTINUOUS SIMULATION OPTIMIZATION

3.1 Introduction

In this chapter, we propose and analyze a framework called Adaptive Search with Resampling and Discarding (ASRD) for continuous simulation optimization. Our ASRD approach improves upon the Adaptive Search with Resampling (ASR) method for continuous simulation optimization proposed by Andradóttir and Prudius [13]. The ASR method samples points from the continuous decision space Θ, decides whether to accept the sampled points (depending on whether they look promising or not), and subsequently ensures that each accepted sampled point has "enough" objective function observations collected at it. It also includes a resampling step aimed at comparing promising points.

However, in continuous decision space, the probability of sampling the same point twice is zero (unless one resamples the same point). Therefore, as the number of iterations grows, the set of sampled and accepted points in ASR keeps growing, and the computational effort of ensuring "enough" objective function observations at each sampled and accepted point could be huge. If some solutions have much better estimated objective function values than other solutions in the set of sampled and accepted solutions, it seems unnecessary to spend effort on obtaining "enough" observations on those inferior points. Motivated by this idea, the ASRD framework developed in this chapter successfully solves the above mentioned drawback of ASR by discarding inferior points that were previously sampled and accepted. Our ASRD

framework also contains a resampling step which is designed for obtaining more precise objective function estimates by collecting additional observations for sampled, accepted, and not discarded points. We discover that whether it is beneficial to resample or not depends on the problem, and we derive an indicator to assist us in deciding whether to resample or not.

This chapter is organized as follows. In Section 3.2, we present our ASRD algorithm, prove its almost sure convergence, discuss the needed assumptions, and propose an efficient acceptance criterion. In Section 3.3, we provide a numerical study aimed at comparing the ASRD and ASR methods and their modifications with each other, and address whether or not it is appropriate to incorporate a resampling procedure in our framework. In Section 3.4, we summarize the main contributions of this chapter.

3.2 Adaptive Search with Resampling and Discarding

We present and analyze our algorithm for continuous simulation optimization. In detail, Section 3.2.1 presents our ASRD method, and Section 3.2.2 proves our algorithm converges to the optimal solution with probability one. Finally, Section 3.2.3 discusses how the assumptions under which our method is guaranteed to converge can be satisfied in practice.

3.2.1 Algorithm Description

This subsection starts by introducing some notation. For all $\theta \in \Theta$ and $k \in \mathbb{N}$, let $N_k(\theta)$ be the number of objective function observations collected at θ by the end of iteration k and let $S_k(\theta)$ be the sum of these $N_k(\theta)$ objective function observations. Also, for all $\theta \in \Theta$ and $k \in \mathbb{N}$, let $\hat{f}_k(\theta) = S_k(\theta)/N_k(\theta)$, and let $f_n(\theta) = \hat{f}_k(\theta)$ be the average of n independent observations of $f(\theta)$. Additional notation definitions are as follows:

Definition 3.2.1. *A sequence $\{a_k\}$ is said to be $O(k^n)$ for some $n \in \mathbb{R}$ if there exists a $C_1 \in \mathbb{R}^+$ such that $0 \leq a_k \leq C_1 k^n$ for all $k \in \mathbb{N}$. A sequence $\{a_k\}$ is said to be*

$\Psi(k^n)$ *for some* $n \in \mathbb{R}$ *if there exists a* $C_2 \in \mathbb{R}^+$ *such that* $a_k \geq C_2 k^n$ *for all* $k \in \mathbb{N}$. *A sequence* $\{a_k\}$ *is said to be* $\Omega(k^n)$ *for some* $n \in \mathbb{R}$ *if it is both* $O(k^n)$ *and* $\Psi(k^n)$.

Next, let $\{K(i)\}_{i=1}^{\infty}$ be a nondecreasing sequence of positive integers, and let $\{V(i)\}_{i=1}^{\infty}$ be a strictly increasing sequence of positive integers with $V(1) = 1$. Let Θ_i be the set of solutions sampled and accepted by the end of iteration $V(i)$ without discarding already accepted points. Let Θ_i^* denote the set of solutions sampled, accepted, and not discarded by the end of iteration $V(i)$. Let Θ_i^+ be the set of solutions sampled and accepted by iteration $V(i)$, and not discarded prior to the discarding procedure in iteration $V(i)$. The pseudo-code for our benchmark ASRD algorithm is given in Algorithm 1.

Algorithm 1 Adaptive Search with Resampling and Discarding

Select $c > 0$, $K(i) = \Psi(i^c)$, $\{\delta_i\}_{i=1}^{\infty}$ a sequence of positive real numbers, a sampling strategy, a resampling strategy, and an acceptance criterion. Let $\Theta_0^* = \emptyset$, $i = 1$, and $k = 0$.
while Stopping criterion is not satisfied **do**
    Let $k = k + 1$
    if $k = V(i)$ **then**
        Sample a solution θ_i from Θ using the sampling strategy
        Based on the acceptance criterion, decide whether to include θ_i in the set Θ_i^+, so that $\Theta_i^+ \in \{\Theta_{i-1}^*, \Theta_{i-1}^* \cup \{\theta_i\}\}$, and update $N_k(\theta_i)$ and $S_k(\theta_i)$ if needed
        For each $\theta \in \Theta_i^+$, if $N_k(\theta) < K(i)$, obtain $K(i) - N_k(\theta)$ additional observations of $f(\theta)$ and update $N_k(\theta)$ and $S_k(\theta)$ accordingly
        Select an estimate of the current best solution $\theta_i^* \in \arg\max_{\theta \in \Theta_i^+} \hat{f}_k(\theta)$
        Let $\Theta_i^* = \Theta_i^+$
        For each $\theta \in \Theta_i^*$, if $\hat{f}_k(\theta_i^*) - \hat{f}_k(\theta) > \delta_i$, remove θ from Θ_i^* and update $\Theta_i^* = \Theta_i^* \backslash \{\theta\}$
        Let $i = i + 1$
    else
        Sample a solution θ from Θ_{i-1}^* using resampling strategy
        Obtain an estimate of $f(\theta)$ and update $N_k(\theta)$ and $S_k(\theta)$
    end if
end while
Return θ_{i-1}^* as an estimate of the optimal solution.

In each iteration of the ASRD method, we either adaptively sample from the feasible region (if the current iteration number is equal to some element in the sequence $\{V(i)\}_{i=1}^{\infty}$), or we resample (if needed) a previously sampled point otherwise. After a new point has been sampled, we decide whether to include the sampled point in the set of sampled, accepted, and not discarded points based on a pre-defined acceptance criterion. The main objective is to include sampled points that appear promising and reject the rest. Simultaneously, we update the number of objective function observations collected at each accepted but not discarded sampled point to grow at least at the rate of $K(i)$, where i is the number of sampled points. Then those points exhibiting inferior qualities are discarded based on a sequence of threshold parameters $\{\delta_i\}_{i=1}^{\infty}$ (i.e., a point θ is discarded if $\hat{f}_{V(i)}(\theta_i^*) - \hat{f}_{V(i)}(\theta) > \delta_i$). A reasonable decision is to let the threshold δ_i decreases as the number of sampling iterations i grows. The intuition is: Originally, the threshold was set to be large due to the noise generated by simulation in the early stages. However, as the number of iterations grows, the noise tends to disappear according to the strong law of large numbers, and the sample average of each point tends to more accurately reflect the true objective function value. As a result, the threshold value is later decreased.

3.2.2 Convergence Analysis

In this section, we present our main convergence result for the ASRD algorithm. For each $\epsilon > 0$, define $\Theta_\epsilon = \{\theta \in \Theta : f(\theta) \geq f^* - \epsilon\}$. For $n \in \mathbb{N}$ and $\theta \in \Theta$, let $f_n(\theta)$ be the estimate of $f(\theta)$ obtained from n observations of $f(\theta)$. Let $\bar{A}$ denote the complement of any set A. Note that *i.o.* stands for "infinitely often" and *a.a.* stands for "almost always." For each $\theta \in \Theta$ and each $\epsilon > 0$, define θ to be a "good" point with respect to ϵ if $f(\theta) \geq f^* - \epsilon$. Otherwise, θ is a "bad" point with respect to ϵ. For any any $x \in \mathbb{R}$, $\lceil x \rceil$ denotes the smallest integer not less than x and $\lfloor x \rfloor$ denotes the largest integer not greater than x.

We also need the following assumptions.

Assumption 3.2.1. *For each $\theta \in \Theta$, if it is of interest to estimate $f(\theta)$, then we generate independent and unbiased observations $\{h(\theta, X_j(\omega)\}_{j=1}^{\infty}$ of $f(\theta)$. Moreover, there exist $l \in \mathbb{N}\backslash\{0,1\}$ and $R \in \mathbb{R}^+$ such that $E[(h(\theta, X(\omega)) - f(\theta))^{2l}] \leq R$ for all $\theta \in \Theta$ and $j \in \mathbb{N}^+$.*

Assumption 3.2.2. *The random elements used for estimating the objective function values (e.g., in steps 7 and 14) are independent of the random elements used in the execution of algorithmic decisions (e.g., in steps 5 and 13).*

Assumption 3.2.3. *For each $\epsilon > 0$, we have $P(\theta_i \in \Theta_i \cap \Theta_\epsilon, i.o.) = 1$.*

Assumption 3.2.1 imposes the finiteness of moments for the random variables under consideration in this chapter. Note that this assumption is weaker than assuming the existence of moment-generating functions in neighborhood of zero, where the latter corresponds to $l = \infty$ in Assumption 3.2.1.

Assumption 3.2.2 imposes restrictions on the random elements used by the ASRD algorithm. It allows for the use of common random numbers to estimate the objective function values $f(\theta)$ at different solutions θ. A similar assumption can be found in Andradóttir and Prudius [13].

The intuition for Assumption 3.2.3 is that since we discard points in our algorithm, we could possibly discard all good points due to estimation error. Hence, we need to be able to find good points again if we discard all of the previously found ones. However, even if Assumption 3.2.3 holds, we will not get almost sure convergence if the estimation is done poorly. This is because we may discard all good points infinitely often, or select a bad point as the estimate of the optimal solution even when good points are available. Therefore, to prove the ASRD algorithm converges almost surely, we need to (i) show that Assumption 3.2.3 holds and (ii) prove almost sure convergence given that Assumption 3.2.3 holds. In this section, we focus on (ii);

(i) is addressed in Section 3.2.3. We first provide an example to illustrate that even when Assumption 3.2.3 holds, if we do estimation poorly, we cannot guarantee the almost sure convergence of the ASRD algorithm.

Example 3.2.1. *Consider the optimization problem* (1) *with:*

$$f(\theta) = \begin{cases} 1 & \text{if } -1 \leq \theta \leq 0, \\ 1.2 & \text{if } 0 < \theta \leq 1, \end{cases}$$

$\Theta = [-1, 1]$, *and*

$$h(\theta, X(\omega)) = \begin{cases} f(\theta) + X(\omega) & \text{if } -1 \leq \theta \leq 0, \\ f(\theta) & \text{if } 0 < \theta \leq 1, \end{cases}$$

where $X(\omega)$ *is a* $\mathcal{N}(0, 100)$ *random variable (*$\mathcal{N}(\mu, \sigma^2)$ *denotes a normal distribution with mean* μ *and variance* σ^2*). The global optimal value is* $f^* = 1.2$*, and the optimal solution set is* $(0, 1]$*. Let* $\Phi(\cdot)$ *denote the cumulative distribution function of* $\mathcal{N}(0, 1)$ *and* $\bar{\Phi}(\cdot) = 1 - \Phi(\cdot)$*.*

We apply the ASRD algorithm as follows: 1) In even sampling steps, we sample a point uniformly from $[-1, 0]$*; in odd sampling steps, we sample a point uniformly from* $(0, 1]$*. We accept every point we sampled. Clearly Assumption 3.2.3 is satisfied. 2) Let* $V(i) = i$*,* $K(i) = 1$*, and* $\delta_i = 0.1$ *for each* $i \in \mathbb{N}^+$*, so that we sample a new point in every step (i.e., there is no resampling), only obtain one objective function observation at each sampled point, and use a constant sequence to discard points.*

In every even sampling step, since we only obtain one objective function observation at each sampled point in the region $[-1, 0]$*, the probability that we obtain a value that is no less than 1.4 is:*

$$P(h(\theta, X(\omega)) \geq 1.4) = P(X(\omega) \geq 0.4) = \bar{\Phi}(0.04) > 0.4.$$

Therefore, with probability at least 0.4*, we obtain an objective function observation that is no less than* 1.4*. Since there is no noise on the optimal solution set and* $\delta_i = 0.1$

for all $i \in \mathbb{N}^+$, we end up discarding all the optimal solutions. In this scenario, the ASRD algorithm does not converge and cannot successfully find the optimal value.

We can see from the above Example 3.2.1 that we need to choose the parameters carefully to guarantee the convergence of the algorithm even under Assumption 3.2.3. In the following, we present our convergence analysis of the ASRD algorithm.

Theorem 3.2.1. *Suppose Assumptions 3.2.1, 3.2.2, and 3.2.3 hold. Let $\delta_i = \frac{D}{i^\gamma}$ for some constants $D > 0$ and $\gamma \geq 0$. If $c(l-1) - 2\gamma l > 2$, then $f(\theta_i^*) \to f^*$ almost surely as $i \to \infty$.*

The theorem condition $c(l-1)-2\gamma l > 2$ implies that given l, we need to choose an appropriate pair of (c, γ) values to make sure the algorithm converges with probability one. For example, if we decide to choose γ to be large, then we need to pick a comparatively large value c to satisfy the theorem condition. The intuition is as follows: if we choose γ to be large, it means the threshold values δ_i decrease rapidly with respect to sampling iteration number i. Therefore, to reduce the likelihood of discarding a good point due to rapid decrease in threshold value, we need to set c large enough to ensure enough data is collected at each sampled, accepted, and not discarded point before executing our discarding procedure. Finally, from the theorem conditions, it is clear that there are no assumptions on the sequence $\{V(i)\}$.

To prove Theorem 3.2.1, we need the following variant of Lemma 1 of Andradóttir and Prudius [13].

Lemma 3.2.1. *Let $\{Z_i\}_{i=1}^{\infty}$ be a sequence of independent random variables with mean zero such that $E[Z_i^{2l}] \leq R < \infty$ for $i, l \in \mathbb{N}^+$. Let $S_n = \sum_{i=1}^{n} Z_i$ for all $n \in \mathbb{N}^+$. Then for each $\epsilon > 0$, we have $P(|S_n| \geq \epsilon n) \leq \frac{l^{2l+1}R}{\epsilon^{2l} n^l}$.*

Proof. Fix $\epsilon > 0$ and $n \geq l$. By Markov's inequality we have that:

$$P(|S_n| \geq \epsilon n) \leq \frac{E[S_n^{2l}]}{(\epsilon n)^{2l}}. \tag{3}$$

According to the proof of Lemma 1 of Andradóttir and Prudius [13], we have

$$E[S_n^{2l}] \leq n^l l^{2l+1} R. \tag{4}$$

Combining (3) and (4) we have the result. □

Below we prove the theorem.

Proof. Fix $\epsilon > 0$. In order to prove the algorithm converges almost surely, it suffices to show the following:

$$P\left(\theta_i^* \in \bar{\Theta}_\epsilon, i.o.\right) = 0.$$

We have:

$$\begin{aligned}
&P\left(\theta_i^* \in \bar{\Theta}_\epsilon, i.o.\right) \\
&\leq P\left(\theta_i^* \in \bar{\Theta}_\epsilon, \Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\right) + P\left(\theta_i^* \in \bar{\Theta}_\epsilon, \Theta_i^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\right) \\
&\leq P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\right) + P\left(\theta_i^* \in \bar{\Theta}_\epsilon, \Theta_i^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\right). \qquad (5)
\end{aligned}$$

It suffices to show that (a) $P(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.) = 0$ and (b) $P(\theta_i^* \in \bar{\Theta}_\epsilon, \Theta_i^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.) = 0$. Note that (a) ensures that all good points are not rejected infinitely often, and (b) ensures that the algorithm does a good job with estimation so that the estimate θ_i^* of the optimal solution is selected well when good points are available.

We first show (a). Since $P(\theta_i \in \Theta_i \cap \Theta_{\epsilon/2}, i.o.) = 1$ from Assumption 3.2.3, we have

$$\begin{aligned}
&P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\right) \\
&= P\left(\{\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\} \cap \{\theta_i \in \Theta_i \cap \Theta_{\epsilon/2}, i.o.\}\right) \\
&\leq P\left(\{\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\} \cap \{\Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\}\right).
\end{aligned}$$

It is not difficult to see that if we sample and accept points within ϵ of the best infinitely often but simultaneously we do not have points within ϵ of the best in our

sampled, accepted, and not discarded solution set infinitely often (event $\{\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\} \cap \{\Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\}$), then it must happen infinitely often that there exist good points with respect to ϵ in the set of sampled, accepted, and not discarded prior to the execution of discarding procedure, and we discard all the good points with respect to ϵ (event $\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.$). Therefore

$$P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\right) \leq P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\right). \tag{6}$$

For each $i \in \mathbb{N}^+$, let $\tilde{\Theta}_i$ be the set of sampled points by the end of iteration $V(i)$. Note $|\tilde{\Theta}_i| \leq i$ (in general $|\tilde{\Theta}| = i$ would be expected, unless the sampling strategy allows for resampling), where $|A|$ denotes the cardinality of set A. Using the methodology of Andradóttir and Prudius [13], suppose that if a sampled point is rejected or discarded, we still collect additional observations at this point to ensure that it has enough observations collected at it (i.e., by the end of iteration $V(i)$ it has at least $K(i)$ observations). Although we collect additional observations at the points in $\tilde{\Theta}_i \backslash \Theta_i^+$, we do not use them for making decisions concerning the evolution of the algorithm. Thus collecting additional data at these points does not impact convergence, and in practice we would not collect this data.

For each $i \in \mathbb{N}^+$, we consider

$$
\begin{aligned}
&P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset\right) \\
&\leq P\left(\bigcup_{\theta\in\Theta_i^+}\bigcup_{\theta'\in\Theta_i^+}\left\{\hat{f}_{V(i)}(\theta') - \hat{f}_{V(i)}(\theta) > \delta_i, \theta' \notin \Theta_{\epsilon/2}, \theta \in \Theta_{\epsilon/2}\right\}\right) \\
&\leq P\left(\bigcup_{\theta\in\tilde{\Theta}_i}\bigcup_{\theta'\in\tilde{\Theta}_i}\left\{\hat{f}_{V(i)}(\theta') - \hat{f}_{V(i)}(\theta) > \delta_i, \theta' \in \bar{\Theta}_{\epsilon/2}, \theta \in \Theta_{\epsilon/2}\right\}\right) \\
&= P\left(\bigcup_{\theta\in\tilde{\Theta}_i}\bigcup_{\theta'\in\tilde{\Theta}_i}\left\{\hat{f}_{V(i)}(\theta') - \hat{f}_{V(i)}(\theta) > \delta_i, f(\theta') < f^* - \epsilon/2, f(\theta) \geq f^* - \epsilon/2\right\}\right) \\
&= P\left(\bigcup_{\theta\in\tilde{\Theta}_i}\bigcup_{\theta'\in\tilde{\Theta}_i}\left\{\left|\hat{f}_{V(i)}(\theta') - f(\theta') + f(\theta) - \hat{f}_{V(i)}(\theta)\right| > \delta_i\right\}\right) \\
&\leq P\left(\bigcup_{\theta\in\tilde{\Theta}_i}\left\{\left|f(\theta) - \hat{f}_{V(i)}(\theta)\right| > \delta_i/2\right\}\right) + P\left(\bigcup_{\theta'\in\tilde{\Theta}_i}\left\{\left|\hat{f}_{V(i)}(\theta') - f(\theta')\right| > \delta_i/2\right\}\right) \\
&\leq 2\sum_{j=1}^{i} P\left(\left|\hat{f}_{V(i)}(\theta_j) - f(\theta_j)\right| > \delta_i/2\right). \qquad (7)
\end{aligned}
$$

For each $\theta_j \in \tilde{\Theta}_i$, let F_j be its law. We have

$$(7) = 2\sum_{j=1}^{i}\int_{\Theta} P\left(|\hat{f}_{V(i)}(\theta_j) - f(\theta_j)| > \delta_i/2 \middle| \theta_j = x_j\right) F_j(\mathrm{d}x_j).$$

Recall that the number of observations collected at each sampled point by the end of iteration $V(i)$ is at least $K(i)$. Thus, for each $j = 1, \ldots, i$, we have:

$$
\begin{aligned}
&P\left(|\hat{f}_{V(i)}(\theta_j) - f(\theta_j)| > \delta_i/2 \middle| \theta_j = x_j\right) \\
&\quad= \sum_{n=K(i)}^{\infty} P\left(|\hat{f}_{V(i)}(x_j) - f(x_j)| > \delta_i/2, N_{V(i)}(x_j) = n \middle| \theta_j = x_j\right) \\
&\quad= \sum_{n=K(i)}^{\infty} P\left(|f_n(x_j) - f(x_j)| > \delta_i/2, N_{V(i)}(x_j) = n \middle| \theta_j = x_j\right) \\
&\quad\leq \sum_{n=K(i)}^{\infty} P\left(|f_n(x_j) - f(x_j)| > \delta_i/2 \middle| \theta_j = x_j\right) \\
&\quad= \sum_{n=K(i)}^{\infty} P\left(|f_n(x_j) - f(x_j)| > \delta_i/2\right). \qquad (8)
\end{aligned}
$$

Equality (8) holds because the objective function observations collected at x_j are independent of the fact that $\theta_j = x_j$.

From Lemma 3.2.1 and Assumption 3.2.1, we know that $P(|f_n(x_j) - f(x_j)| > \frac{\delta_i}{2}) \leq \frac{2^{2l} l^{2l+1} R}{\delta_i^{2l} n^l}$. Therefore:

$$(8) \leq \sum_{n=K(i)}^{\infty} \frac{Const}{\delta_i^{2l} n^l}, \tag{9}$$

where $Const$ is some constant.

Plugging the bounds $(8) - (9)$ into the original formula (7) yields:

$$P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset\right) \leq 2\left(\sum_{j=1}^{i} \int_{\Theta} \sum_{n=K(i)}^{\infty} \frac{Const}{\delta_i^{2l} n^l} F_j(\mathrm{d}x_j)\right) = 2\sum_{j=1}^{i} \sum_{n=K(i)}^{\infty} \frac{Const}{\delta_i^{2l} n^l}. \tag{10}$$

Recalling that $K(i) = \Psi(i^c)$, we have $\exists\ C_1 > 0, I \geq 0$, s.t. $K(i) - 1 \geq C_1 i^c$ for $i > I$. Therefore, for $i > I$

$$\sum_{n=K(i)}^{\infty} \frac{Const}{\delta_i^{2l} n^l} \leq \int_{K(i)-1}^{\infty} \frac{Const}{\delta_i^{2l} s^l} \mathrm{d}s \leq \int_{C_1 i^c}^{\infty} \frac{Const}{\delta_i^{2l} s^l} \mathrm{d}s = \frac{C_2}{\delta_i^{2l} i^{c(l-1)}}, \tag{11}$$

where $C_2 = \frac{Const}{(l-1)C_1^{(l-1)}}$ is a positive constant (recall that $l \geq 2$ from Assumption 3.2.1). By inflating C_2 if necessary, (11) also holds for $i \leq I$. Hence

$$(10) \leq 2\sum_{j=1}^{i} \frac{C_2}{\delta_i^{2l} i^{c(l-1)}} = \frac{2C_2 i}{\delta_i^{2l} i^{c(l-1)}} = \frac{2C_2}{\delta_i^{2l} i^{c(l-1)-1}}. \tag{12}$$

From the theorem conditions, we know that $\delta_i = \frac{D}{i^\gamma}$. From $(10) - (12)$, we have:

$$P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset\right) \leq \frac{2C_2}{D^{2l} i^{c(l-1)-1-2\gamma l}}. \tag{13}$$

Therefore

$$\sum_{i=1}^{\infty} P\left(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2} \neq \emptyset\right) \leq \sum_{i=1}^{\infty} \frac{2C_2}{D^{2l} i^{c(l-1)-1-2\gamma l}} < \infty$$

since $c(l-1) - 1 - 2\gamma l > 1$ from the theorem assumption. According to (6) and the first Borel-Cantelli lemma, we have $P(\Theta_i^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.) = 0$. Hence we have proved (a).

In the following, we need to show (b). Consider

$$P(\theta_i^* \in \bar{\Theta}_\epsilon, \Theta_i^* \cap \Theta_{\epsilon/2} \neq \emptyset) \leq P\left(\bigcup_{\theta \in \Theta_i^*} \left\{|\hat{f}_{V(i)}(\theta) - f(\theta)| \geq \epsilon/4\right\}\right)$$
$$\leq P\left(\bigcup_{\theta \in \tilde{\Theta}_i} \left\{|\hat{f}_{V(i)}(\theta) - f(\theta)| \geq \epsilon/4\right\}\right).$$

Using the same methodology as in $(7)-(12)$, we can bound the above probability by $\frac{Const}{i^{c(l-1)-1}}$ for all i, where $Const$ is a positive constant. Therefore, $\sum_{i=1}^{\infty} P(\theta_i^* \in \bar{\Theta}_\epsilon, \Theta_i^* \cap \Theta_{\epsilon/2} \neq \emptyset) \leq \sum_{i=1}^{\infty} \frac{C}{i^{c(l-1)-1}}$. From the theorem assumption, we have $c(l-1) - 2\gamma l > 2$, which implies that $c(l-1) - 1 > 1$. Thus $\sum_{i=1}^{\infty} \frac{C}{i^{c(l-1)-1}} < \infty$. Again, applying the first Borel-Cantelli lemma, we know that $P(\theta_i^* \in \bar{\Theta}_\epsilon, \Theta_i^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.) = 0$. This completes the proof. □

Remark 3.2.1. *The original ASR method of Andradóttir and Prudius [13] updates the estimate of the optimal solution in each iteration and is guaranteed to converge almost surely when* $V(i) = \lfloor i^b \rfloor$ *for all* i, $b \geq 1$, $c(l-1) > b+1$, *and mild conditions similar to Assumptions* 3.2.1, 3.2.2, *and* 3.2.3 *(without discarding) hold. From the proof of statement (b) in Theorem 3.2.1, we know that under Assumptions* 3.2.1, 3.2.2, *and* 3.2.3, *ASR will converge almost surely when* $c(l-1) > 2$ *and the estimate of the optimal solution is updated only when a new point is sampled.*

In Theorem 3.2.1, if $\gamma > 0$, the sequence $\{\delta_i\}$ is strictly decreasing in i and goes to 0 as i goes to ∞. On the other hand, if $\gamma = 0$, we have a constant positive sequence $\{\delta_i\}$. Motivated by this property, we have the following Theorem 3.2.2, which indicates that any positive sequence $\{\delta_i\}$ that is bounded away from zero guarantees the almost sure convergence of the ASRD algorithm. Note that the $\delta_i = \delta$ for all $i \in \mathbb{N}^+$ case of Theorem 3.2.2 corresponds to the $\gamma = 0$ case of Theorem 3.2.1.

Theorem 3.2.2. *Suppose Assumptions 3.2.1, 3.2.2, and 3.2.3 hold. Let* $\{\delta_i\}$ *be a sequence of positive numbers such that* $\inf_i \delta_i \geq \delta$ *for some* $\delta > 0$. *If* $c(l-1) > 2$, *then* $f(\theta_i^*) \to f^*$ *almost surely as* $i \to \infty$.

Proof. From the proof of Theorem 3.2.1, we know that to show the algorithm convergences almost surely, it is sufficient to show both $\sum_{i=1}^{\infty} \frac{1}{\delta_i^{2l} i^{c(l-1)-1}}$ and $\sum_{i=1}^{\infty} \frac{1}{i^{c(l-1)-1}}$ are finite. Since $c(l-1)-1 > 1$ from the theorem assumption, we have $\sum_{i=1}^{\infty} \frac{1}{i^{c(l-1)-1}} < \infty$, implying that $\sum_{i=1}^{\infty} \frac{1}{\delta_i^{2l} i^{c(l-1)-1}} < \infty$ since $\inf_i \delta_i \geq \delta$ for some $\delta > 0$. □

From Theorems 3.2.1 and 3.2.2, we can see that we need a stronger condition to guarantee the almost surely convergence of the algorithm if we allow the sequence $\{\delta_i\}$ to go to 0 as i goes to ∞. This is reasonable because the discarding procedure is more likely to make mistakes in discarding points, and hence we need to choose c, l and γ with more discretion.

Looking back at the ASRD algorithm, we only update the estimate of the optimal solution in iterations $V(i)$ for $i \in \mathbb{N}^+$. We now prove the almost sure convergence of a modified version of the ASRD algorithm where the optimal solution is updated at the end of each iteration.

Theorem 3.2.3. *Suppose Assumptions 3.2.1, 3.2.2, and 3.2.3 hold. Consider the ASRD algorithm with the following modifications: We add one more step after step 14, which is to obtain the current estimate of the optimal solution θ_k^*, we denote the estimate of the optimal solution in steps 8 and 10 as θ_k^* instead of θ_i^*, and we replace θ_{i-1}^* by θ_k^* in step 17. Also let $V(i) = \lfloor i^b \rfloor$ for $i \in \mathbb{N}^+$ and $b \geq 1$. We have:*

(I) If $\delta_i = \frac{D}{i^\gamma}$ for some constants $D > 0$, $\gamma \geq 0$, and each $i \in \mathbb{N}^+$, $c(l-1)-2\gamma l > 2$, and $\frac{c(l-1)-1}{b} > 1$, then $f(\theta_k^*) \to f^*$ almost surely as $k \to \infty$.

(II) If $\{\delta_i\}$ is a sequence of positive numbers such that $\inf_i \delta_i \geq \delta$ for some $\delta > 0$ and $\frac{c(l-1)-1}{b} > 1$, then $f(\theta_k^*) \to f^*$ almost surely as $k \to \infty$.

Proof. As in the proof of Theorem 3.2.1, we will prove that for each $\epsilon > 0$,

$$P\left(\theta_k^* \in \bar{\Theta}_\epsilon, i.o.\right) = 0.$$

Let $m_k = \lfloor k^{1/b} \rfloor$. Note that m_k is the number of points sampled by the end of iteration

k. We have:

$$
\begin{aligned}
&P\left(\theta_k^* \in \bar{\Theta}_\epsilon, i.o.\right) \\
&\leq P\left(\theta_k^* \in \bar{\Theta}_\epsilon, \Theta_{m_k}^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\right) + P\left(\theta_k^* \in \bar{\Theta}_\epsilon, \Theta_{m_k}^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\right) \\
&\leq P\left(\Theta_{m_k}^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.\right) + P\left(\theta_k^* \in \bar{\Theta}_\epsilon, \Theta_{m_k}^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\right) \qquad (14)
\end{aligned}
$$

According to the proofs of Theorems 3.2.1 and 3.2.2, we have $P(\Theta_{m_k}^* \cap \Theta_{\epsilon/2} = \emptyset, i.o.) = 0$ when either $c(l-1) - 2\gamma l > 2$ and $\delta_i = \frac{D}{i^\gamma}$ for some constants $D > 0$ and $\gamma \geq 0$, or $c(l-1) > 2$ and δ_i is a sequence of positive numbers such that $\inf_i \delta_i \geq \delta$ for some $\delta > 0$, respectively. Note that $c(l-1) - 1 \geq \frac{c(l-1)-1}{b} > 1$ in the second case since $b \geq 1$.

We also need to show $P\left(\theta_k^* \in \bar{\Theta}_\epsilon, \Theta_{m_k}^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\right)$. Consider

$$
\begin{aligned}
P\left(\theta_k^* \in \bar{\Theta}_\epsilon, \Theta_{m_k}^* \cap \Theta_{\epsilon/2} \neq \emptyset\right) &\leq P\left(\bigcup_{\theta \in \Theta_{m_k}^*} \left\{|\hat{f}_k(\theta) - f(\theta)| \geq \epsilon/4\right\}\right) \\
&\leq P\left(\bigcup_{\theta \in \tilde{\Theta}_{m_k}} \left\{|\hat{f}_k(\theta) - f(\theta)| \geq \epsilon/4\right\}\right).
\end{aligned}
$$

Using the same method as in $(7) - (12)$ but with i replaced by m_k, we can bound the above probability by $\frac{Const}{k^{\frac{c(l-1)-1}{b}}}$ for all k, where $Const$ is some constant. Therefore

$$
\sum_{k=1}^{\infty} P\left(\theta_k^* \in \bar{\Theta}_\epsilon, \Theta_{m_k}^* \cap \Theta_{\epsilon/2} \neq \emptyset\right) \leq \sum_{k=1}^{\infty} \frac{Const}{k^{\frac{c(l-1)-1}{b}}} < \infty.
$$

Applying the first Borel-Cantelli lemma, we know that

$$
P\left(\theta_k^* \in \bar{\Theta}_\epsilon, \Theta_{m_k}^* \cap \Theta_{\epsilon/2} \neq \emptyset, i.o.\right) = 0.
$$

This completes the proof. □

We now compare the results of Theorem 3.2.1 and part (I) of Theorem 3.2.3. From the conditions of Theorem 3.2.3, if $V(i) = \lfloor i^b \rfloor$ for all i, where $b \geq 1$, and $2\gamma l + 1 \geq b$, then $c(l-1) - 2\gamma l > 2$ implies $\frac{c(l-1)-1}{b} > 1$. In this case, estimating the optimal

solution in each iteration will not impact convergence. However, if $2\gamma l + 1 < b$, then we need more conditions to make sure the ASRD algorithm converges almost surely if we estimate optimal solution in each iteration. One reasonable explanation is that if l is not large enough to satisfy the condition $2\gamma l + 1 \geq b$, then the random variable $h(\theta, X(\omega))$ has high volatility and we do not sample and discard points often (because b is not small and γ is not large). Hence we cannot rely on the solutions extracted from steps other than $V(i)$. A similar result holds for Theorem 3.2.2 and part (II) of Theorem 3.2.3.

We conclude this section by showing that if we choose the sequence $\{\delta_i\}$ to be decreasing and convergent to 0 as i goes to ∞, we can make sure every sampled and accepted bad point with respect to ϵ will be discarded eventually for any $\epsilon > 0$ after carefully choosing c and l. The theorem is as follows:

Theorem 3.2.4. *Suppose Assumptions 3.2.1, 3.2.2, and 3.2.3 hold. Let $\{\delta_i\}$ be a sequence of positive numbers such that $\delta_i \to 0$ as $i \to \infty$. If $c(l-1) > 1$, then for any $\epsilon > 0$, every point not within ϵ of the best will eventually get discarded. Explicitly, we have*

$$P\left(\bigcup_{i=1}^{\infty}\left\{\theta_i \in \bigcap_{j\geq i}\Theta_j^*, \theta_i \in \bar{\Theta}_\epsilon\right\}\right) = 0.$$

Proof. We have

$$P\left(\bigcup_{i=1}^{\infty}\left\{\theta_i \in \bigcap_{j\geq i}\Theta_j^*, \theta_i \in \bar{\Theta}_\epsilon\right\}\right) \leq \sum_{i=1}^{\infty} P\left(\theta_i \in \bigcap_{j\geq i}\Theta_j^*, \theta_i \in \bar{\Theta}_\epsilon\right). \tag{15}$$

For each $i \geq 1$, we have

$$
\begin{aligned}
& P\left(\theta_i \in \bigcap_{j \geq i} \Theta_j^*, \theta_i \in \bar{\Theta}_\epsilon\right) \\
& \leq P\left(\left(\bigcap_{j=i+1}^{\infty}\left(\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_j \in \Theta_j^+\right\} \bigcup \{\theta_j \notin \Theta_j^+\}\right)\right) \bigcap \{\theta_i \in \Theta_i^* \cap \bar{\Theta}_\epsilon\}\right) \\
& \leq P\left(\left(\bigcap_{j=i+1}^{\infty}\left(\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j\right\} \bigcup \{\theta_j \notin \Theta_j^+\}\right)\right) \bigcap \{\theta_i \in \bar{\Theta}_\epsilon\}\right) \\
& \leq P\left(\bigcap_{j=i+1}^{\infty}\left(\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_i \in \bar{\Theta}_\epsilon\right\} \bigcup \{\theta_j \notin \Theta_j^+\}\right)\right). \qquad (16)
\end{aligned}
$$

From Assumption 3.2.3, we know that $P\left(\bigcap_{k=1}^{\infty} \bigcup_{j=k}^{\infty}\left\{\theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\right\}\right)=1$. (Notice that for each j, the event $\{\theta_j \in \Theta_j \cap \Theta_{\epsilon/2}\}$, where θ_j is the new point sampled in iteration j, is equivalent to the event $\{\theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\}$.) Hence we have

$$
\begin{aligned}
& (16) \\
& = P\left(\left(\bigcap_{j=i+1}^{\infty}\left(\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_i \in \bar{\Theta}_\epsilon\right\} \bigcup \{\theta_j \notin \Theta_j^+\}\right)\right) \bigcap \left(\bigcap_{k=1}^{\infty} \bigcup_{j=k}^{\infty}\{\theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\}\right)\right). \qquad (17)
\end{aligned}
$$

Moreover, for any $\tau \geq i$, $\tau \in \mathbb{N}^+$,

$$
\begin{aligned}
& (17) \\
& \leq P\left(\left(\bigcap_{j=\tau+1}^{\infty}\left(\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_i \in \bar{\Theta}_\epsilon\right\} \bigcup \{\theta_j \notin \Theta_j^+\}\right)\right) \bigcap \left(\bigcap_{k=1}^{\infty} \bigcup_{j=k}^{\infty}\{\theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\}\right)\right) \\
& \leq P\left(\left(\bigcap_{j=\tau+1}^{\infty}\left(\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_i \in \bar{\Theta}_\epsilon\right\} \bigcup \{\theta_j \notin \Theta_j^+\}\right)\right) \bigcap \left(\bigcup_{j=\tau+1}^{\infty}\{\theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\}\right)\right) \\
& \leq P\left(\bigcup_{j=\tau+1}^{\infty}\left(\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_i \in \bar{\Theta}_\epsilon, \theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\right\} \bigcup \{\theta_j \notin \Theta_j^+, \theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\}\right)\right). \qquad (18)
\end{aligned}
$$

As $\left\{\theta_j \notin \Theta_j^+, \theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\right\}$ is an empty set for each j, we have

$$
\begin{aligned}
(18) &= P\left(\bigcup_{j=\tau+1}^{\infty}\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_i \in \bar{\Theta}_\epsilon, \theta_j \in \Theta_j^+ \cap \Theta_{\epsilon/2}\right\}\right) \\
&\leq P\left(\bigcup_{j=\tau+1}^{\infty}\left\{\hat{f}_{V(j)}(\theta_j) \leq \hat{f}_{V(j)}(\theta_i)+\delta_j, \theta_i \in \bar{\Theta}_\epsilon, \theta_j \in \Theta_{\epsilon/2}\right\}\right). \qquad (19)
\end{aligned}
$$

Since δ_i goes to 0 as i goes to ∞, for any $\epsilon > 0$ we let $N(\epsilon)$ be the number such that $2\delta_j < \epsilon$ for all $j > N(\epsilon)$. Without loss of generality, we choose $\tau \geq N(\epsilon)$, then we have

$$
\begin{aligned}
(19) &\leq P\left(\bigcup_{j=\tau+1}^{\infty}\left\{|f(\theta_j) - \hat{f}_{V(j)}(\theta_j) + \hat{f}_{V(j)}(\theta_i) - f(\theta_i)| \geq \epsilon/2 - \delta_j\right\}\right) \\
&\leq P\left(\bigcup_{j=\tau+1}^{\infty}\left(\left\{|f(\theta_j) - \hat{f}_{V(j)}(\theta_j)| \geq \frac{\epsilon - 2\delta_j}{4}\right\}\bigcup\left\{|\hat{f}_{V(j)}(\theta_i) - f(\theta_i)| \geq \frac{\epsilon - 2\delta_j}{4}\right\}\right)\right) \\
&\leq \sum_{j=\tau+1}^{\infty}\left[P\left(|f(\theta_j) - \hat{f}_{V(j)}(\theta_j)| \geq \frac{\epsilon - 2\delta_j}{4}\right) + P\left(|\hat{f}_{V(j)}(\theta_i) - f(\theta_i)| \geq \frac{\epsilon - 2\delta_j}{4}\right)\right] \\
&\leq \sum_{j=\tau+1}^{\infty}\int_{\Theta} P\left(|f(\theta_j) - \hat{f}_{V(j)}(\theta_j)| \geq \frac{\epsilon - 2\delta_j}{4}\middle|\, \theta_j = x_j\right) F_j(\mathrm{d}x_j) \\
&\qquad + \sum_{j=\tau+1}^{\infty}\int_{\Theta} P\left(|\hat{f}_{V(j)}(\theta_i) - f(\theta_i)| \geq \frac{\epsilon - 2\delta_j}{4}\middle|\, \theta_i = x_i\right) F_i(\mathrm{d}x_i), \qquad (20)
\end{aligned}
$$

where F_i and F_j denote the laws of θ_i and θ_j, respectively.

Using exactly the same technique as in the proof of Theorem 3.2.1 (see (8), (9), and (11)), we have

$$
P\left(|\hat{f}_{V(j)}(\theta_j) - f(\theta_j)| \geq \frac{\epsilon - 2\delta_j}{4}\middle|\, \theta_j = x_j\right) \leq \frac{Const}{(\epsilon - 2\delta_j)^{2l} j^{c(l-1)}}
$$

and

$$
P\left(|\hat{f}_{V(j)}(\theta_i) - f(\theta_i)| \geq \frac{\epsilon - 2\delta_j}{4}\middle|\, \theta_i = x_i\right) \leq \frac{Const}{(\epsilon - 2\delta_j)^{2l} j^{c(l-1)}},
$$

where $Const$ denotes some constant. Recalling that the sequence $\{\delta_j\}$ is decreasing and goes to 0, we have $\delta_j \leq \delta_{N(\epsilon)}$ for $j \geq \tau + 1$. Thus we have

$$
\frac{Const}{(\epsilon - 2\delta_j)^{2l} j^{c(l-1)}} \leq \frac{Const}{(\epsilon - 2\delta_{N(\epsilon)})^{2l} j^{c(l-1)}}. \qquad (21)
$$

Therefore (16) – (21) imply

$$
\begin{aligned}
P\left(\theta_i \in \bigcap_{j \geq i} \Theta_j^*, \theta_i \in \bar{\Theta}_\epsilon\right) &\leq \sum_{j=\tau+1}^{\infty}\left[\int_{\Theta} \frac{Const}{(\epsilon - 2\delta_{N(\epsilon)})^{2l} j^{c(l-1)}} F_j(\mathrm{d}x_j) + \int_{\Theta} \frac{Const}{(\epsilon - 2\delta_{N(\epsilon)})^{2l} j^{c(l-1)}} F_i(\mathrm{d}x_i)\right] \\
&= \sum_{j=\tau+1}^{\infty} \frac{Const}{(\epsilon - 2\delta_{N(\epsilon)})^{2l} j^{c(l-1)}}. \qquad (22)
\end{aligned}
$$

We know that (22) holds for any $\tau \geq \max\{N(\epsilon), i\}$ and we also have $c(l-1) > 1$ from the theorem assumption. Hence we can let $\tau \to \infty$, yielding

$$\lim_{\tau\to\infty} \sum_{j=\tau+1}^{\infty} \frac{Const}{(\epsilon - 2\delta_{N(\epsilon)})^{2l} j^{c(l-1)}} = 0.$$

Therefore we can conclude $P\left(\theta_i \in \bigcap_{j\geq i} \Theta_j^*, \theta_i \in \bar{\Theta}_\epsilon\right) = 0$. Since i is arbitrary, the result now follows from (15). □

Remark 3.2.2. *From the proof of Theorem 3.2.4, we can tell that if the sequence* $\{\delta_i\}$ *does not converge to* 0*, we cannot get the result for each* $\epsilon > 0$*. However, if* ϵ *is fixed and* $\sup_i \delta_i < \epsilon/2$*, the result of Theorem 3.2.4 holds, and every bad point with respect to* ϵ *will be discarded.*

3.2.3 Discussion of Assumption 3.2.3

In this section, we use two specific sampling strategies and an acceptance criterion to verify Assumption 3.2.3 as stated in Section 3.2.2. Let $\mathcal{G}$ be a collection of distributions with index set $\mathcal{I}$. We will need the following assumption.

Assumption 3.2.4. *For each* $\epsilon > 0$*, we have* $\inf_{i\in\mathcal{I}} G_i(\Theta_\epsilon) > 0$.

Under the above assumption, we first choose the following sampling strategy denoted as $[RS]$ for Random Search: At each sampling step, choose a distribution G_i from $\mathcal{G}$ and then sample a point using distribution G_i independent of everything.

Proposition 3.2.1. *Under Assumption 3.2.4, if sampling strategy* $[RS]$ *is used and we accept every sampled point, then Assumption 3.2.3 holds.*

Proof. Let $\epsilon > 0$. In each sampling step i, the probability of sampling a good point is $G_i(\Theta_\epsilon)$ for some $G_i \in \mathcal{G}$, regardless of the past information. As we accept every sampled point, we have that $P(\theta_i \in \Theta_i \cap \Theta_\epsilon) \geq \inf_{j\in\mathcal{I}} G_j(\Theta_\epsilon) > 0$ from Assumption 3.2.4. Therefore:

$$\sum_{i=1}^{\infty} P(\theta_i \in \Theta_i \cap \Theta_\epsilon) = \infty.$$

From the second Borel-Cantelli lemma, we know Assumption 3.2.3 holds. This completes the proof. □

Next, we modify the sampling strategy as follows (the new sampling strategy is denoted $[AS]$ for Adaptive Search): Assume there is a family of local sampling distributions $\mathcal{L}$ aiming at searching promising subregions based on the current information we have. For example, it can be a local search around the current best solution. At any iteration i ($i > 1$), with probability $0 < p \leq 1$, we sample a point using distribution G_i from $\mathcal{G}$ independent of everything and with probability $1 - p$, we sample a point using a local distribution L_i chosen from $\mathcal{L}$.

Proposition 3.2.2. *Under Assumption 3.2.4, if sampling strategy $[AS]$ is used and we accept every sampled point, then Assumption 3.2.3 holds.*

Proof. Let $\epsilon > 0$. Since we accept every point we sample, we only need to prove that the sampling strategy $[AS]$ samples good points infinitely often. We have that $P(\theta_i \in \Theta_i \cap \Theta_\epsilon) \geq p \inf_{j \in \mathcal{I}} G_j(\Theta_\epsilon)$. Since $\inf_{j \in \mathcal{I}} G_j(\Theta_\epsilon) > 0$ from Assumption 3.2.4 and $p > 0$, therefore,

$$\sum_{i=1}^{\infty} P(\theta_i \in \Theta_i \cap \Theta_\epsilon) = \infty.$$

Again, from the second Borel-Cantelli lemma, we know Assumption 3.2.3 holds. This completes the proof. □

We next describe the acceptance criterion [AH] (for Andradóttir and Hu) that is used in our ASRD algorithm. Let $\lambda > 0$ and let $\{H(i)\}$ be a sequence of positive integers. We obtain $H(i)$ independent observations of $f(\theta_i)$ after we sample a new point. The newly sampled solution θ_i is included in the set Θ_i^+ of sampled, accepted and not discarded points in iteration i if an objective function estimate based on $H(i)$ observations at this point is at least as good as the estimated objective function value at the best solution found so far minus an indifference parameter λ. Explicitly, if $\hat{f}_{V(i-1)}(\theta_{i-1}^*) - f_{H(i)}(\theta_i) \leq \lambda$, then accept the sampled point θ_i, otherwise, reject

this point. Note that in Andradóttir and Prudius [13], the acceptance criterion [AP] (for Andradóttir and Prudius) is: Let $\lambda > 0$, the newly sampled solution is included in the set Θ_k of sampled and accepted points in iteration k if an objective function estimate based on $K \geq 1$ observations at this point is at least as good as the estimated objective function value at the best solution found so far minus an indifference parameter λ (i.e., a sampled point θ is accepted if $f_K(\theta) \geq \hat{f}^*_{k-1}(\theta_{k-1}) - \lambda$. The main difference between acceptance criteria [AH] and [AP] is that a time-varying number of function observations are collected at the newly sampled solution in [AH] whereas a constant number of function observations are collected at the newly sampled solution in [AP]. The main advantage of [AH] over [AP] is the former approach is more flexible (numerical results in Section 3 shows [AH] is more efficient than [AP]).

Proposition 3.2.3. *Suppose Assumptions 3.2.1 and 3.2.2 hold,* $0 < \epsilon < \lambda$*, we choose* $\delta_i = \frac{D}{i^\gamma}$ *for some constants* $D > 0$ *and* $\gamma \geq 0$*, select* c *to satisfy* $c(l-1) > 2$*, and we sample good points with respect to* ϵ *infinitely often with probability one. If acceptance criterion [AH] is used and* $H(i) = \lceil Qi^q \rceil$ *for all* i*, where* Q *and* q *are positive real numbers satisfying* $ql > 1$*, then Assumption 3.2.3 holds.*

Proof. We need to prove that $P(\theta_i \in \Theta_i \cap \Theta_\epsilon, i.o.) = 1$ or equivalently $P(\{\theta_i \in \Theta_i \cap \Theta_\epsilon, i.o.\}^c) = 0$. Since we sample good points infinitely often with probability one, we have that $P(\theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon, i.o.) = 1$ (recall that $\tilde{\Theta}_i$ is the set of sampled points by the end of iteration $V(i)$). Therefore, we have:

$$\begin{aligned} P\left(\{\theta_i \in \Theta_i \cap \Theta_\epsilon, i.o.\}^c\right) &= P\left(\{\theta_i \in \Theta_i \cap \Theta_\epsilon, i.o.\}^c \bigcap \{\theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon, i.o.\}\right) \\ &= P\left(\{\theta_i \notin \Theta_i \cap \Theta_\epsilon, a.a.\} \bigcap \{\theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon, i.o.\}\right) \\ &\leq P\left(\{\theta_i \notin \Theta_i, \theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon, i.o.\}\right). \end{aligned} \tag{23}$$

To prove that (23) equals zero, let $i > 1$ (we do not consider $i = 1$ is because we

always accept the first sampled point). For each i:

$$
\begin{aligned}
P\left(\theta_i \notin \Theta_i, \theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon\right) &= P\left(\left\{\bigcup_{\theta\in\Theta^*_{i-1}} \left\{\hat{f}_{V(i-1)}(\theta) - f_{H(i)}(\theta_i) > \lambda\right\}\right\} \bigcap \left\{\theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon\right\}\right) \\
&= P\left(\bigcup_{\theta\in\Theta^*_{i-1}} \left\{\hat{f}_{V(i-1)}(\theta) - f_{H(i)}(\theta_i) > \lambda, \theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon\right\}\right) \\
&\leq P\left(\bigcup_{\theta\in\Theta^*_{i-1}} \left\{\hat{f}_{V(i-1)}(\theta) - f(\theta) + f(\theta_i) - f_{H(i)}(\theta_i) > \lambda - \epsilon\right\}\right). \qquad (24)
\end{aligned}
$$

Inequality (24) is due to the fact that $f(\theta_i) \geq f^* - \epsilon$ and $f(\theta) \leq f^*$, and therefore we have $f(\theta_i) - f(\theta) \geq -\epsilon$. Since $\lambda > \epsilon$ from our assumption and $\Theta^*_{i-1} \subseteq \tilde{\Theta}_{i-1}$, we have

$$
\begin{aligned}
(24) &\leq P\left(\bigcup_{\theta\in\tilde{\Theta}_{i-1}} \left\{|\hat{f}_{V(i-1)}(\theta) - f(\theta) + f(\theta_i) - f_{H(i)}(\theta_i)| > \lambda - \epsilon\right\}\right) \\
&\leq P\left(\bigcup_{\theta\in\tilde{\Theta}_{i-1}} \left\{|\hat{f}_{V(i-1)}(\theta) - f(\theta)| > \frac{\lambda-\epsilon}{2}\right\}\right) + P\left(\bigcup_{\theta\in\tilde{\Theta}_{i-1}} \left\{\left|f(\theta_i) - f_{H(i)}(\theta_i)\right| > \frac{\lambda-\epsilon}{2}\right\}\right) \\
&= P\left(\bigcup_{\theta\in\tilde{\Theta}_{i-1}} \left\{|\hat{f}_{V(i-1)}(\theta) - f(\theta)| > \frac{\lambda-\epsilon}{2}\right\}\right) + P\left(\left|f(\theta_i) - f_{H(i)}(\theta_i)\right| > \frac{\lambda-\epsilon}{2}\right). \qquad (25)
\end{aligned}
$$

Using exactly the same method and reasoning as in $(7)-(8)$, we can bound the above probability as follows:

$$
\begin{aligned}
(25) &\leq \sum_{j=1}^{i-1} \int_\Theta P\left(|\hat{f}_{V(i-1)}(\theta_j) - f(\theta_j)| > \frac{\lambda-\epsilon}{2} \middle| \theta_j = x_j\right) F_j(\mathrm{d}x_j) \\
&\quad + \int_\Theta P\left(|f(\theta_i) - f_{H(i)}(\theta_i)| > \frac{\lambda-\epsilon}{2} \middle| \theta_i = x_i\right) F_i(\mathrm{d}x_i) \\
&= \sum_{j=1}^{i-1} \int_\Theta \sum_{n=K(i-1)}^{\infty} P\left(|f_n(x_j) - f(x_j)| > \frac{\lambda-\epsilon}{2}\right) F_j(\mathrm{d}x_j) \\
&\quad + \int_\Theta P\left(\left|f(x_i) - f_{H(i)}(x_i)\right| > \frac{\lambda-\epsilon}{2}\right) F_i(\mathrm{d}x_i). \qquad (26)
\end{aligned}
$$

From (9), (11), and Lemma 3.2.1, we have $\sum_{n=K(i-1)}^{\infty} P\left(|f_n(x_j) - f(x_j)| > \frac{\lambda-\epsilon}{2}\right) \leq \frac{Const}{(i-1)^{c(l-1)}}$ and $P\left(\left|f(x_i) - f_{H(i)}(x_i)\right| > \frac{\lambda-\epsilon}{2}\right) \leq \frac{Const}{i^{ql}}$, where $Const$ is some positive constant. Therefore $(24)-(26)$ yield that for each $i > 0$, we have

$$
P\left(\theta_i \notin \Theta_i, \theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon\right) \leq \frac{Const}{(i-1)^{c(l-1)-1}} + \frac{Const}{i^{ql}}. \qquad (27)
$$

Since $c(l-1)-1>1$ and $ql>1$ from the assumptions in the lemma, we have

$$\sum_{i=1}^{\infty} P\left(\theta_i \notin \Theta_i, \theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon\right) < \infty.$$

From the first Borel-Cantelli lemma, we know that $P\left(\{\theta_i \notin \Theta_i, \theta_i \in \tilde{\Theta}_i \cap \Theta_\epsilon, i.o.\}\right) = 0$. Therefore, (23) yields $P\left(\{\theta_i \in \Theta_i \cap \Theta_\epsilon, i.o.\}^c\right) = 0$. This completes the proof. □

Remark 3.2.3. *Propositions 3.2.1 and 3.2.2 show that under Assumption 3.2.4, both sampling strategies [RS] and [AS] sample good points with respect to $\epsilon > 0$ infinitely often. Therefore, from Proposition 3.2.3, we know that if [RS] is combined with [AH], or [AS] is combined with [AH], then Assumption 3.2.3 is satisfied.*

3.3 Numerical Examples

The main contribution of ASRD is the incorporation of an efficient discarding procedure. In addition, a more flexible acceptance criterion is proposed. Therefore, in this section, we compare the ASRD algorithm developed in this chapter with the ASR method of Andradóttir and Prudius [13] (that does not have discarding), implemented both with the original acceptance criterion of [13] and with our new acceptance criterion [AH]. Moreover, to further investigate the effects of resampling on the performances of the algorithms, we also test these methods without resampling. The notation we use to denote the algorithms is given in Table 1.

The outline of this section is as follows: In Section 3.3.1, we describe our test problems, in Section 3.3.2, we provide implementation details for the tested algorithms, and in Section 3.3.3, we compare the numerical performance of these methods. Finally, in Section 3.3.4, we provide suggestions on identifying whether resampling is necessary or not in the algorithms.

3.3.1 Test Problems

This section describes our test problems. The following five benchmark problems, which have been previously studied, e.g., in Andradóttir and Prudius [13], and Hu,

Table 1: Notation

Notation	Algorithm
ASRD[AH]	Adaptive Search with Resampling and Discarding, implemented with acceptance criterion [AH] and $\delta_i = \frac{D}{i^\gamma}$ for $i \in \mathbb{N}^+$
ASRD[AP]	Adaptive Search with Resampling and Discarding, implemented with acceptance criterion [AP] and $\delta_i = \frac{D}{i^\gamma}$ for $i \in \mathbb{N}^+$
ASD[AH]	ASRD[AH] without resampling
ASD[AP]	ASRD[AP] without resampling
ASR[AH]	Adaptive Search with Resampling of Andradóttir and Prudius [13] implemented with acceptance criterion [AH]
ASR[AP]	Adaptive Search with Resampling of Andradóttir and Prudius [13]
AS[AH]	ASR[AH] without resampling implemented with acceptance criterion [AH]
AS[AP]	ASR[AP] without resampling

Fu and Marcus [38], are used in our experiments. The first two are low-dimensional problems that have simple structures. The third is a 10-dimensional highly multimodal problem. The fourth is a 20-dimensional, badly scaled problem. Finally, the fifth is a 20-dimensional, highly multimodal problem. For the third and fifth problems, the number of local optima increases exponentially with the problem dimension.

The Smooth problem:

$$f(\theta) = -[(x_1 - 0.5)\sin(10x_1) + (x_2 + 0.5)\cos(5x_2)],$$

$\Theta = \{(x_1, x_2) \subseteq \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 1\}$, and for each $\theta \in \Theta$, $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ and $X(\omega)$ is a $\mathcal{N}(0, 1)$ random variable. The approximate range of the objective function values is $(-3, 1.502]$. The optimal value is $f^* \simeq 1.502$.

The Two Hills problem:

$$f(\theta) = \max\{f_1(\theta), f_2(\theta), 0\},$$

where $f_1(\theta) = -(0.4x_1 - 5)^2 - 2(0.4x_2 - 17.2)^2 + 7$ and $f_2(\theta) = -(0.4x_1 - 12)^2 - (0.4x_2 - 4)^2 + 4$. The feasible region is given by $\Theta = \{(x_1, x_2) \in \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 50\}$. We let $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ for all $\theta \in \Theta$, as for the smooth problem, with $X(\omega)$ being $\mathcal{N}(0, 100)$ for all $\theta \in \Theta$. This objective function is of interest, mainly, because it has two hills of different heights (4 and 7), located relatively far apart (the

hill of height 4 is centered at $(30, 10)$ and the hill of height 7 is centered at $(12.5, 43)$), and separated by a flat valley (of height 0). Notice that the standard deviation of the white noise is greater than the range of the objective function values (the range is $[0, 7]$). This makes the problem relatively difficult to solve. The optimal value is $f^* = 7$.

The Pinter 10D problem:

$$f(\theta) = -\left(\sum_{i=1}^{s} i x_i^2 + \sum_{i=1}^{s} i \sin^2\left(x_{i-1}\sin x_i - x_i + \sin x_{i+1}\right)\right) - \left(\sum_{i=1}^{s} i \log_{10}\left[1 + i(x_{i-1}^2 - 2x_i + 3x_{i+1} - \cos x_i + 1)^2\right]\right) - 1,$$

where $x_0 = x_s$, $x_{s+1} = x_1$, and $s = 10$. The feasible region is

$$\Theta = \left\{(x_1, \ldots, x_s) \in \mathbb{R}^s : -10 \leq x_i \leq 10, i = 1, \ldots, s\right\}.$$

The form of $h(\theta, X(\omega))$ is as for the other two test problems, with $X(\omega)$ being $\mathcal{N}(0, 100)$ for all $\theta \in \Theta$. The approximate range is $(-10000, -1]$, and this problem has a global maximum at $(0, \ldots, 0)$ and $f^* = -1$.

The Rosenbrock 20D problem:

$$f(\theta) = -\left(\sum_{i=1}^{s-1}\left[(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2\right] + 1\right),$$

where $s = 20$. The feasible region is

$$\Theta = \left\{(x_1, \ldots, x_s) \in \mathbb{R}^s : -10 \leq x_i \leq 10, i = 1, \ldots, s\right\}.$$

The form of $h(\theta, X(\omega))$ is as for the other three test problems, with $X(\omega)$ being $\mathcal{N}(0, 100)$ for all $\theta \in \Theta$. Note that this problem is highly volatile with the approximate range of the objective function values being $(-10^8, -1]$. It has a global maximum at $(1, \ldots, 1)$ and $f^* = -1$.

The Griewank 20D problem:

$$f(\theta) = -\left(\frac{1}{4}\sum_{i=1}^{s} x_i^2 - \prod_{i=1}^{s}\cos(\frac{x_i}{\sqrt{i}}) + 2\right),$$

where $s = 20$. The feasible region is

$$\Theta = \{(x_1, \ldots, x_s) \in \mathbb{R}^s : -10 \leq x_i \leq 10, i = 1, \ldots, s\}.$$

The form of $h(\theta, X(\omega))$ is as for the other four test problems, with $X(\omega)$ being $\mathcal{N}(0, 100)$ for all $\theta \in \Theta$. The approximate range is $(-600, -1]$, and this problem has a global maximum at $(0, \ldots, 0)$ and $f^* = -1$.

3.3.2 Algorithm Implementation

This section provides implementation details for the ASRD[AH], ASRD[AP], ASD[AH], ASD[AP], ASR[AH], ASR[AP], AS[AH], and AS[AP] approaches. In order to compare the performance of these algorithms, we use the same algorithm parameter values and the same sampling and resampling strategies. Moreover, our sampling and resampling strategies agree with those of Andradóttir and Prudius [13].

We first describe the sampling and resampling strategies. The sampling procedure, which is a special case of adaptive search, is as follows. In iteration $k = V(i)$, with probability $p > 0$, a new solution is sampled uniformly from the whole feasible set Θ, and with probability $1 - p$, a new solution is sampled uniformly from $N(\theta^*_{i-1})$, where

$$N(\theta) = N((x_1, \ldots, x_s)) = \{(x'_1, \ldots, x'_s) \in \Theta : |x_i - x'_i| \leq r, i = 1, \ldots, s\}$$

for all $\theta \in \Theta$ (the first point is sampled uniformly from Θ). Here we use r denote the radius of the "local" neighborhood.

The resampling procedure in iteration k is as follows: Let $V(i) = \lfloor i^b \rfloor$, where $b \geq 1$, and note that $m_k = \lfloor k^{1/b} \rfloor$ is the number of points sampled by the end of iteration k. Then, a point $\theta \in \Theta^*_{m_k}$ is resampled in iteration k with probability

$$p_k(\theta) = \frac{\exp\{\hat{F}_{m_k}(\theta)\}}{\sum_{\theta' \in \Theta^*_{m_k}} \exp\{\hat{F}_{m_k}(\theta')\}},$$

where $\hat{F}_{m_k}(\theta) = \min\{\max\{\underline{U}, \hat{f}_{m_k}(\theta)/T\}, \overline{U}\}$, with $\overline{U} > \underline{U} > 0$ and $T > 0$. The reason we have the bounds $\overline{U}$ and $\underline{U}$ is to restrict $\hat{F}_{m_k}(\theta)$ in a reasonable range, which makes

$\exp\{\hat{F}_{m_k}(\theta)\}$ neither too big nor too small for the computer to calculate. This resampling procedure puts more weight on the points that have better estimated objective function values. Moreover, we only update the resampling probability measure when the set $\Theta^*_{m_k}$ may have changed (i.e., when a new point is sampled and a number of old points may have been discarded).

We would like to mention that since our framework has an efficient procedure to discard inferior points, and since the resampling procedure is aimed at focusing sampling on superior points, it is possible that resampling is no longer necessary. However, from the numerical examples in Section 3.3.3, we will see that when we only have a limited simulation budget, whether or not to conduct resampling in the algorithm is still a case by case issue.

Next we describe the acceptance criterion for the algorithms. In the ASRD[AH], ASD[AH], ASR[AH], and AS[AH] methods, the newly sampled point θ is accepted if $\hat{f}_{V(i-1)}(\theta^*_{i-1}) - f_{H(i)}(\theta) \leq \lambda$, where $\{H(i)\}$ is the sequence described in Proposition 3.2.3. In the ASRD[AP], ASD[AP], ASR[AP], and AS[AP] methods, the newly sampled point θ is accepted if $\hat{f}_{k-1}(\theta^*_{k-1}) - f_K(\theta) \leq \lambda$, where $K \in \mathbb{N}^+$ is constant. Last, we let $K(i) = \lceil Ci^c \rceil$, where $c, C > 0$.

The parameter values for ASR[AP] and AS[AP] are the same as in Andradóttir and Prudius [13]. Regarding the ASRD[AH], ASRD[AP], ASD[AH], ASD[AP], ASR[AH], and AS[AH] algorithms, we use the same parameter values as ASR[AP] for the sampling and resampling procedures. For the acceptance and discarding criteria, we choose the parameters based on the total computational budgets, the range of the objective function values, and trial and error. For a given problem, how to determine a priori the most appropriate values of these parameters, especially the sequence $\{\delta_i\}$, is still an open issue and outside the scope of this thesis.

In our test problems, the parameter values for ASR[AP] are $b = 1.1$, $c = 0.5$, $C = 1$, $p = 0.5$, $\lambda = 0.01$, and $K = 10$, with r being 0.02 for the Smooth problem,

1.0 for the Two Hills problem, and 0.4 for Pintér 10D problem, Rosenbrock 20D problem, and Griewank 20D problem. Here we choose r to be $\frac{1}{50}$ of the diameter of the feasible set for each problem, where the diameter is defined using infinite norm. In the resampling procedure, let $T = 0.1$ for the Smooth and Two Hills problems, and $T = 1$ for the Pinter 10D, Rosenbrock 20D, and Griewank 20D problems. We take 5 more samples for each point we choose to resample in each step. The additional parameter values for ASRD[AH] are $Q = 1$, $q = 0.05$, $\gamma = 0.2$, $D = 1$ for the Smooth problem, and $D = 10$ for the Two Hills problem, the Pintér 10D problem, the Rosenbrock 20D, and the Griewank 20D problem. Also, we use $\gamma > 0$ instead of $\gamma = 0$ to discard as many bad points as possible. Finally, we choose D as the standard deviation of the noise (in practice this value would of course need to be estimated), $\overline{U} = 400$, and $\underline{U} = -400$. The other algorithms use the same parameter values.

Let $N_k = \sum_{\theta \in \tilde{\Theta}_{m_k}} N_k(\theta)$ be the total number of objective function evaluations by the end of iteration k. Let N be the simulation budget. The performance of the algorithms is averaged over 100 independent replications for all test problems. Their performance is documented by plotting 100 pairs (x, y), where $x \in \{0.01N, 0.02N, \ldots, N\}$, and y is the average objective function value at the estimated optimal solution after x objective function observations have been collected. As the estimate of the optimal solution is only updated in iterations $V(1)$, $V(2)$,..., the value of y is the same for all corresponding $x \in [N_{V(i)}, N_{V(i+1)})$.

3.3.3 Algorithm Comparison

Figures $1-5$ show the empirical performance of the methods described in Table 1 on the Smooth, Two Hills, Pintér 10D, Rosenbrock 20D, and Griewank 20D problems, respectively. For the Smooth and Two Hills problems, we plot $f(\theta_k^*)$ for ASR[AH], ASR[AP], AS[AH], and AS[AP], and we plot $f(\theta_{m_k}^*)$ for ASRD[AH], ASRD[AP],

ASD[AH], and ASD[AP] (recall that ASR updates the optimal solution in each iteration, whereas ASRD only does so in sampling iterations). Similarly, we plot $-f(\theta_k^*)$ for the Pintér 10D, Rosenbrock 20D, and Griewank 20D problems (as all three functions have negative objective function values); therefore smaller values are better for Figures 3, 4 and 5 (larger values are better for Figures 1 and 2). Also, for the Pinter's 10D and Rosenbrock 20D problems, we plot $-f(\theta_k^*)$ on a logarithmic scale (rather than on a linear scale) to facilitate comparisons as the simulation effort increases. Finally, the sequence in which the numerical results are presented moves from lower dimensional, smoother problems to higher dimensional problems with greater curvature. In the following, we will analyze these five problems in detail.

For the Smooth problem (Figure 1), we can see clearly that the resampling procedure significantly improves the overall performance of ASRD[AH], ASRD[AP], ASR[AH], and ASR[AP] compared to ASD[AH], ASD[AP], AS[AH], and AS[AP]. We see that without resamping, algorithms with discarding perform much better, and acceptance criterion [AH] outperforms [AP] as well. When resampling is incorporated, we can see that ASRD[AH], ASRD[AP], ASR[AH], and ASR[AP] have similar performance in later stage of the simulation, however ASRD[AH] and ASR[AH] outperform ASRD[AP] and ASR[AP] at the early stage. One explanation is that in this low-dimensional, smooth problem, discarding has less impact on performance given a good acceptance criterion.

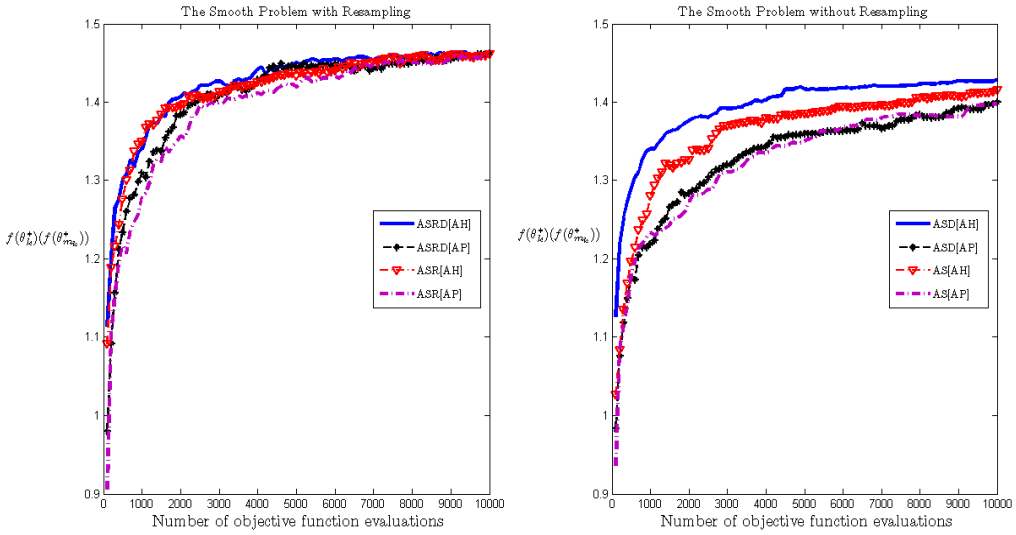

Figure 1: Performance of the optimization methods on the Smooth problem

For the Two Hills problem (Figure 2), without the resampling procedure, it is evident that ASD[AH] has the best performance among all other algorithms implemented and is noticeably better than the second best AS[AH], especially at the early stage of the simulation process. Hence, the acceptance criterion [AH] is able to work with the discarding procedure to expedite the convergence of our framework. Discarding allows us to focus on better points early on when the number of objective function evaluations is small. For the algorithms with resampling, ASRD[AH] and ASR[AH] have slightly better performance than the other two. The reasons are similar as for the smooth problem.

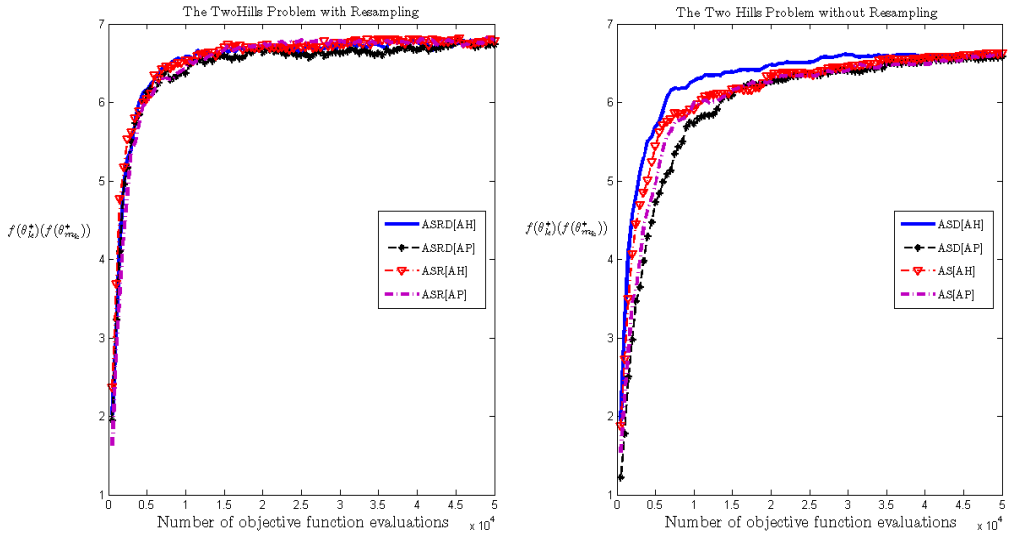

Figure 2: Performance of the optimization methods on the Two Hills problem

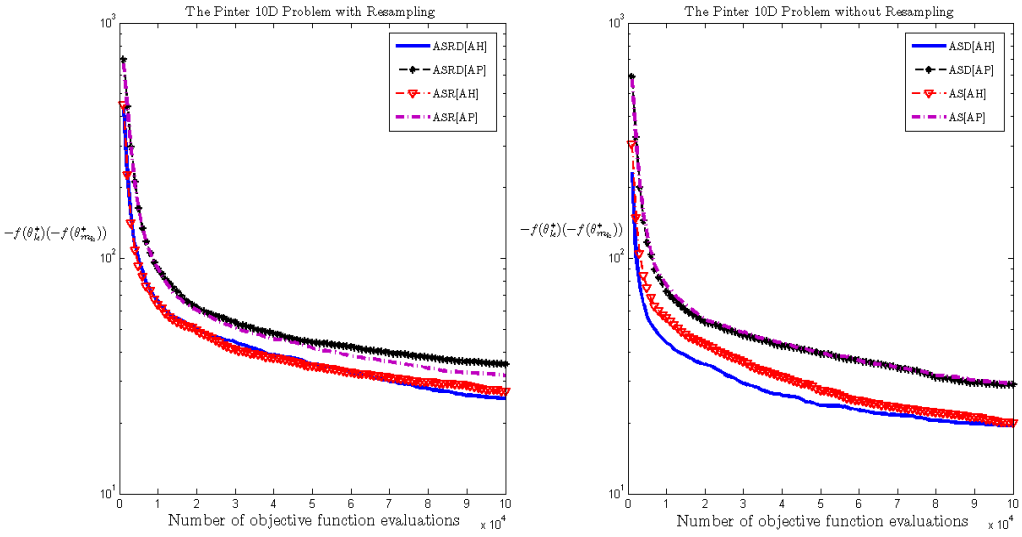

Figure 3: Performance of the optimization methods on the Pinter 10D problem

Next, we look at the high dimensional problems. For the Pinter 10D problem (Figure 3), ASD[AH] and AS[AH] outperform the other algorithms by a large margin, ASD[AH] performs better than AS[AH] by a noticeable margin, and ASRD[AH] has

similar performance as ASR[AH]. The resampling procedure has a negative effect on the performance of the algorithms (especially the ones employing our acceptance criterion [AH]). The reason is the objective function values for the Pintér 10D problem can be as small as around -10^4 and as large as -1. Therefore, the noise, $\mathcal{N}(0, 100)$, is negligible compared to the difference between objective function values. As a result, resampling is redundant and a waste of simulation budget when the noise is negligible. Finally, we can see that discarding helps more compared to the lower-dimensional problems. One explanation is that the Pinter 10D problem has a lot of local optima, and discarding efficiently removes inferior local optima.

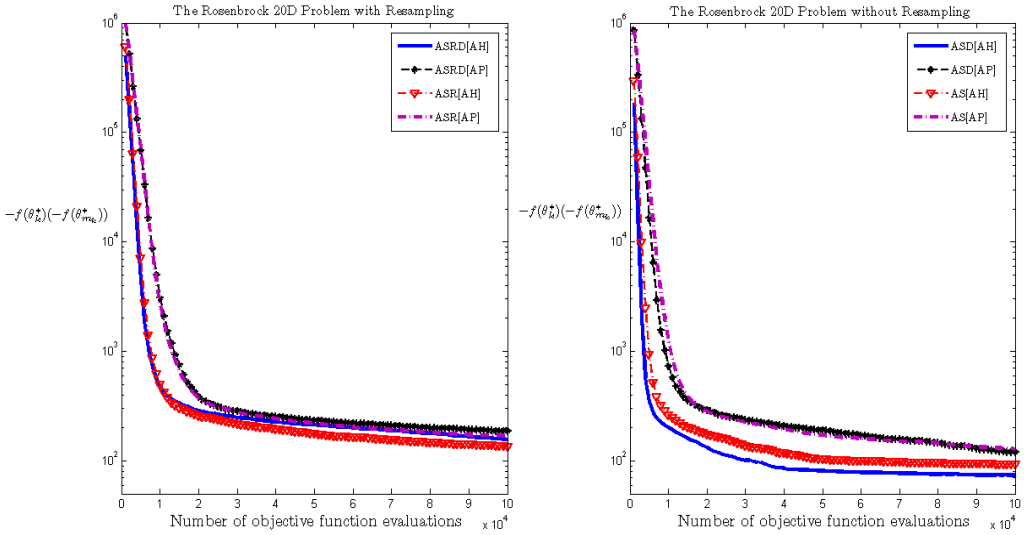

Figure 4: Performance of the optimization methods on the Rosenbrock 20D problem

For the Rosenbrock 20D problem (Figure 4), ASD[AH] has better performance than the other five algorithms, and by a large margin (recall that we plot the objective function values on a logarithmic scale). In addition, we see that ASD[AH] and AS[AH] perform much better than the other algorithms, and that resampling is detrimental on this problem. One contributor (similar to the Pinter 10D problem) is that the Rosenbrock 20D problem itself is highly volatile, i.e., the objective function values

can be as small as approximately -10^8 and as large as -1, whereas the noise is $\mathcal{N}(0, 100)$, which is negligible in most cases unless the sampled points are very close to the optimal solution. Hence the resampling procedure wastes effort on reducing negligible noise. Therefore, an efficient acceptance criterion is very important in this situation, thus the performances of ASD[AH] and AS[AH] exceed the rest by a large margin. Finally, we see that ASD[AH] performs noticeably better than AS[AH] due to the fact that the discarding procedure saves computation efforts on inferior points, enabling the algorithm to focus on points with better potential objective function estimates.

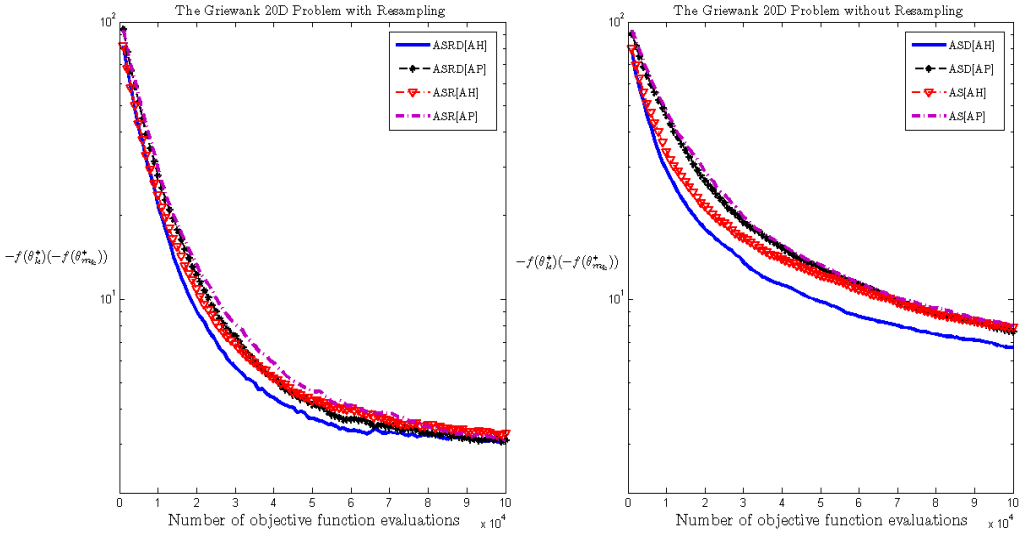

Figure 5: Performance of the optimization methods on the Griewank 20D problem

For the Griewank 20D problem (Figure 5), we can see that ASRD[AH] performs better than all other algorithms throughout the simulation process. In this problem, since the noise is not negligible compared to the range of objective function values, the resampling procedure plays a vital role in achieving better objective function values. Moreover, our test problem is a highly multimodal problem that has a lot of local optima and the values of those local optima are close. Therefore, efficiently

discarding inferior points is very important, as it not only helps our framework to achieve good objective function evaluations at the early stage, but also saves a lot of computational effort updating inferior points compared to the algorithms without the discarding procedure.

In summary, from Figures 1 to 5, we observe:

- When resampling is desirable, ASRD[AH] performs best.
- Acceptance criterion [AH] performs better than [AP] in most cases.
- Either ASRD[AH] or ASD[AH] performs best among all the algorithms; whether ASRD[AH] is better than ASD[AH] or not depends on the noise.

Furthermore, in low dimensional problems (the Smooth and Two Hills problems), discarding helps less if the acceptance criterion is good enough and the performance difference becomes small as the number of simulation iterations grows. However, in high dimensional problems (the Pintér 10D, Rosenbrock 20D, and Griewank 20D problems), discarding makes noticeable progress in addition to a good acceptance criterion. Thus discarding seems to help more when the underlying optimization problem is difficult, which is of course the important case to consider.

3.3.4 Assessing the Desirability of Resampling

The numerical analysis in Section 3.3.3 illustrates that the discarding procedure improves performance in all the test problems. However, whether to implement the resampling procedure is less clear. For test problems 1, 2, and 5, resampling helps the framework work better, whereas resampling hurts the performance of our framework on test problems 3 and 4. The purpose of the resampling procedure is to reduce the noise to get more accurate estimates of the objective function values, and, more explicitly, given any $\theta_i, \theta_j \in \Theta$ with $f(\theta_i) > f(\theta_j)$, resampling helps to reduce the

possibility of the event:

$$E = \{\hat{f}(\theta_i) \leq \hat{f}(\theta_j)\},$$

where $\hat{f}(\cdot)$ denotes an estimate of $f(\cdot)$. However, if event E itself is a rare event under any circumstance, resampling becomes redundant and wastes computational resources that can be used to seek better points. In order to estimate the probability of event E, for any $\theta_i, \theta_j \in \Theta$ satisfying $f(\theta_i) = E[h(\theta_i, X(\omega)] > E[h(\theta_j, X(\omega))] = f(\theta_j)$, we apply the Markov and Cauchy-Schwartz inequalities to bound the following probability:

$$\begin{aligned}
&P\Big(h(\theta_i, X(\omega)) \leq h(\theta_j, X(\omega))\Big) \\
&= P\Big((h(\theta_j, X(\omega)) - f(\theta_j)) - (h(\theta_i, X(\omega)) - f(\theta_i)) \geq f(\theta_i) - f(\theta_j)\Big) \\
&\leq P\Big(\,|(h(\theta_j, X(\omega)) - f(\theta_j)) - (h(\theta_i, X(\omega)) - f(\theta_i))| \geq f(\theta_i) - f(\theta_j)\Big) \\
&\leq \frac{E\left[|(h(\theta_j, X(\omega)) - f(\theta_j)) - (h(\theta_i, X(\omega)) - f(\theta_i))|\right]}{f(\theta_i) - f(\theta_j)} \\
&\leq \frac{[Var(h(\theta_i, X(\omega))) + Var(h(\theta_j, X(\omega))) - 2Cov(h(\theta_i, X(\omega)), h(\theta_j, X(\omega)))]^{\frac{1}{2}}}{f(\theta_i) - f(\theta_j)} \\
&\leq \frac{[Var(h(\theta_i, X(\omega))) + Var(h(\theta_j, X(\omega))) + 2|Cov(h(\theta_i, X(\omega)), h(\theta_j, X(\omega)))|]^{\frac{1}{2}}}{f(\theta_i) - f(\theta_j)} \\
&\leq \frac{\left[Var(h(\theta_i, X(\omega))) + Var(h(\theta_j, X(\omega))) + 2\sqrt{Var(h(\theta_i, X(\omega)))Var(h(\theta_j, X(\omega)))}\right]^{\frac{1}{2}}}{f(\theta_i) - f(\theta_j)} \\
&= \frac{\sqrt{Var(h(\theta_i, X(\omega)))} + \sqrt{Var(h(\theta_j, X(\omega)))}}{f(\theta_i) - f(\theta_j)}. \qquad (28)
\end{aligned}$$

The above bound (28) provides a reasonable indicator to determine whether the resampling procedure is beneficial. If the value of the bound is very small, i.e., close to 0, it suggests that the event E is a rare event and resampling is a wasteful use of simulation time. If the value is large, it suggests that the noise is not a negligible factor when decisions are made. As a result the resampling procedure is needed to obtain more precise evaluations of the objective function values.

Next, we describe a pre-processing procedure we implemented on all five test problems. The details are as follows: While the total number of runs is less than the simulation pre-processing budget, which is defined as B, in each iteration i, we sample a point θ_i via pure random search in Θ, and then we collect L independent objective

function observations at the sampled point θ_i (L is a fixed integer and greater than one). We estimate $E[h(\theta_i, X(\omega))]$ and $Var(h(\theta_i, X(\omega)))$ through their unbiased estimators $f_L(\theta_i)$ and $\frac{1}{L-1}\sum_{j=1}^{L}[f^{(j)}(\theta_i)-f_L(\theta_i)]^2$, respectively, where $f_L(\theta_i) = \frac{\sum_{j=1}^{L} f^{(j)}(\theta_i)}{L}$ and $f^{(j)}(\theta_i)$ denotes the jth collected observation of the objective function value at θ_i.

Let $\underline{\theta}_i$ denote the point with the worst estimated objective function value when i points have been sampled. To utilize the upper bound (28) to determine whether resampling is necessary or not, we document the performance by plotting ten pairs (x, y), where

$$x \in \{0.1B, 0.2B, \ldots, B\}$$

and

$$y = \frac{|f_L(\theta^*_{i_x}) - f_L(\underline{\theta}_{i_x})|}{\sqrt{\frac{1}{L-1}\sum_{i=1}^{L}[f^{(i)}(\theta^*_{i_x}) - f_L(\theta^*_{i_x})]^2} + \sqrt{\frac{1}{L-1}\sum_{i=1}^{L}[f^{(i)}(\underline{\theta}_{i_x}) - f_L(\underline{\theta}_{i_x})]^2}},$$

where $i_x = \lfloor x/L \rfloor$ (recall that θ^*_i denotes the point with the best estimated objective function value when i points have been sampled). Notice that given x, y is the reciprocal of the estimate of the probability bound (28) with the points $\theta^*_{i_x}$ and $\underline{\theta}_{i_x}$. We use the quantity y to estimate the scale of objective function value improvement versus the scale of the noise. The parameters values are $L = 20$ and $B = 2000$. The bounds we obtain are averaged over 100 independent replications of each problem, respectively. Figures 6 and 7 show ten (x, y) pairs for all five test problems. The higher the value of y, the lower the probability bound is, which is equivalent to saying that the resampling procedure is more likely to be redundant.

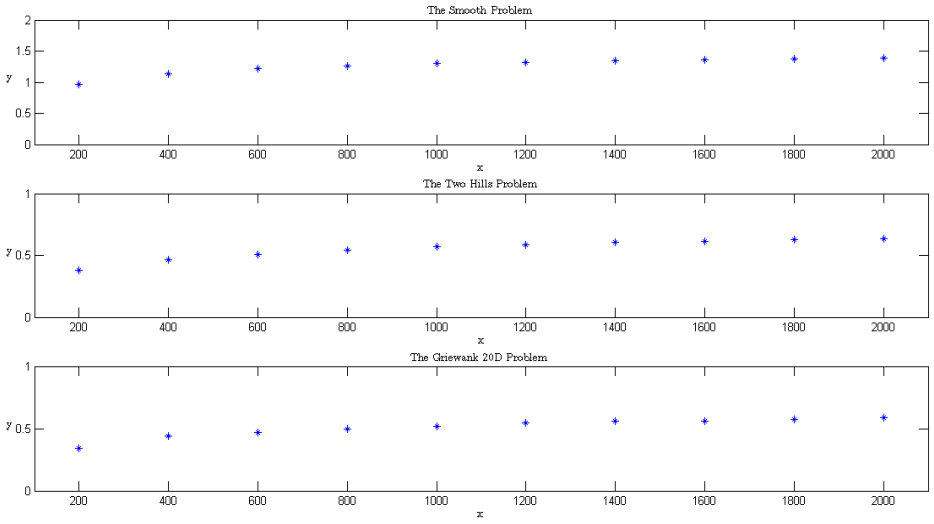

Figure 6: The reciprocal of the estimate of the bounds (28) on test problems 1, 2, and 5

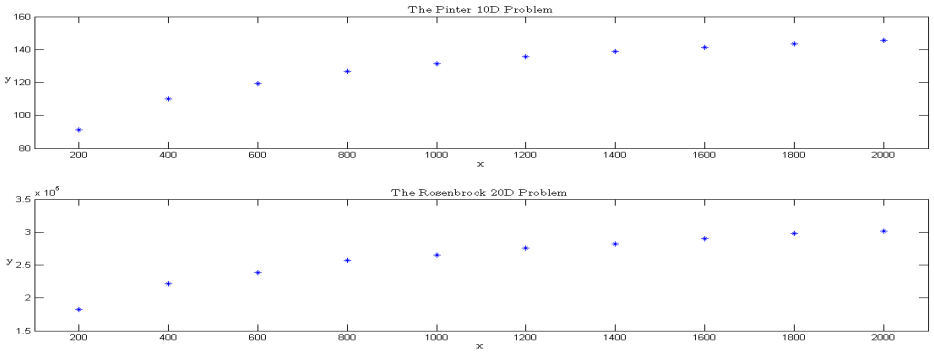

Figure 7: The reciprocal of the estimate of the bounds (28) on test problem 3 and 4

The data in Figure 6 shows that for the Smooth, Two Hills, and Griewank 20D problems, the reciprocals of the estimates of the bound (28) are small, which suggests we need the resampling procedure to reduce chance of the event E happening. This result is consistent with our numerical results in Figures 1, 2, and 5 in Section 3.3. The data in Figure 7 suggests that the event E is a rare event for the Pinter 10D and

Rosenbrock 20D problems, which suggests that resampling is detrimental. Again, the result is consistent with the simulation outcome in Figures 3 and 4 in Section 3.3.3.

Motivated by the fact that in the smooth, two hills, and Griewank 20D problems, the range of the objective function values and the standard deviation of the noise are similar, whereas in the Pinter 10D and Rosenbrock 20D problems, the objective function values have much larger range than the standard deviation of the noise, we conduct additional numerical experiments for the Pinter 10D and Rosenbrock 20D problems with higher noise to understand the relationship between resampling and estimation noise more comprehensively. Since the the approximate ranges for the Pinter 10D and the Rosenbrock 20D problems are $(-10^4, -1]$ and $(-10^8, -1]$ respectively, we set the noise to be $\mathcal{N}(0, 10^6)$ for the Pinter 10D problem, and $\mathcal{N}(0, 10^{10})$ for the Rosenbrock 20D problem. We set D to be the standard deviation of the noise for each test problem, and all other parameter values are the same as in Section 3.3.2. The results are shown in Figures 8 and 9.

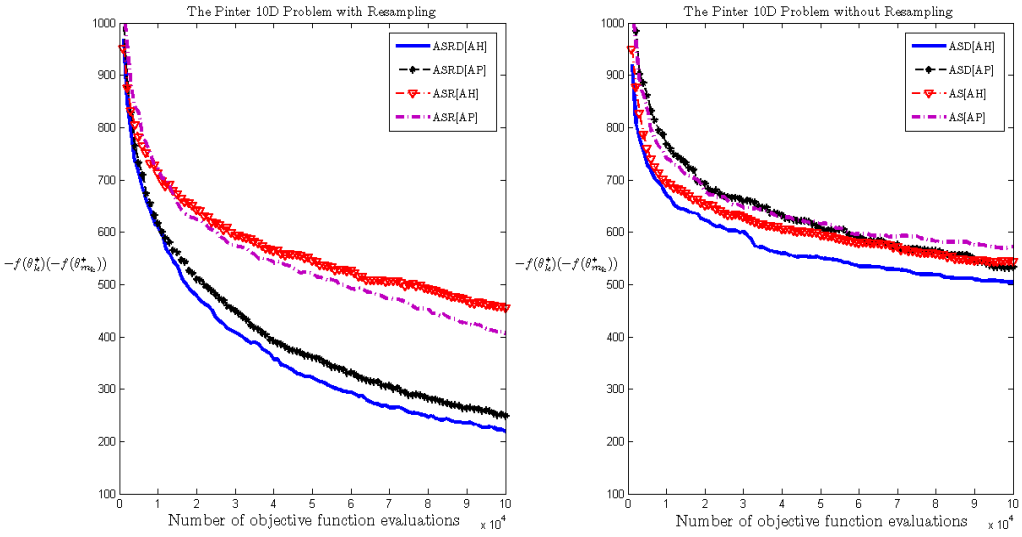

Figure 8: Performance of the optimization methods on the Pinter 10D problem with noise $\mathcal{N}(0, 10^6)$

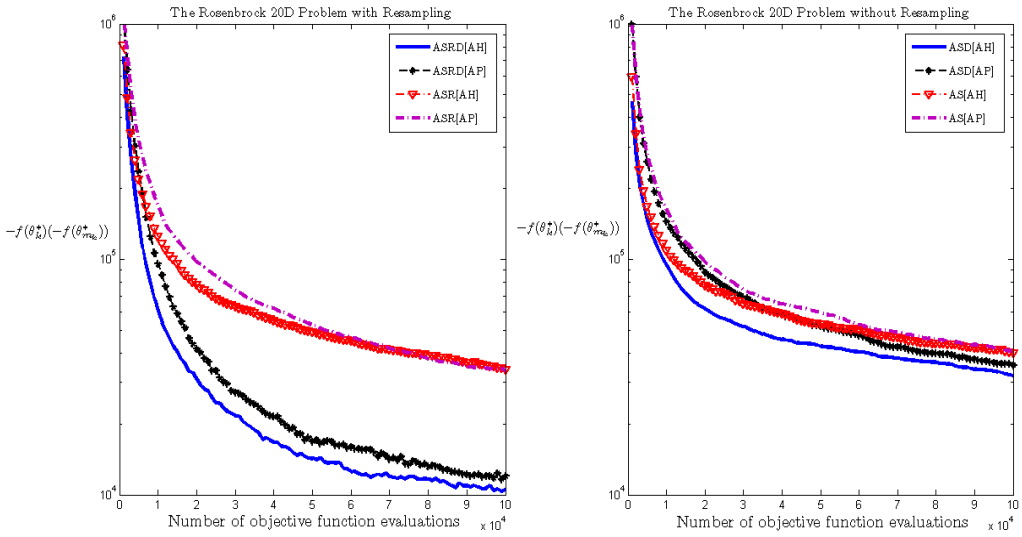

Figure 9: Performance of the optimization methods on the Rosenbrock 20D problem with noise $\mathcal{N}(0, 10^{10})$

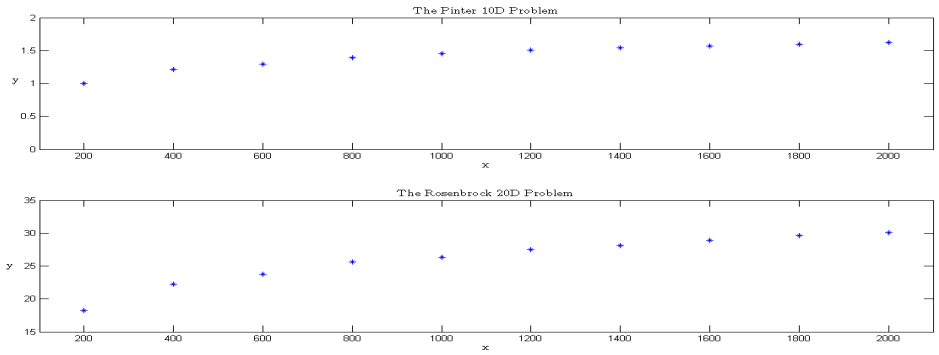

Figure 10: The reciprocal of the estimate of the bounds (28) on test problems 3 and 4 with high noise

For the Pinter 10D problem with high noise (Figure 8), ASRD[AH] and ASRD[AP] outperform the other algorithms by a large margin and resampling greatly improves the performance of the algorithms. This is understandable since the high noise, $\mathcal{N}(0, 10^6)$, cannot be ignored, and resampling helps to reduce the additional noise and

is necessary when the noise is big. We also notice that in Figure 8, ASR[AH] performs better than ASR[AP] at the early stage but worse later. One possible explanation is that the number of function observations in our time-varying discarding procedure [AH] grows too slowly in our experiment. Thus ASR[AH] saves expending simulation effort on bad points early on, but may yield mistakes (due to high noise), especially later on as the estimate of the optimal objective function value improves.

For the Rosenbrock 20D problem with high noise (Figure 9), we notice that ASRD[AH] performs much better than the other algorithms, with ASRD[AP] also performing well (second best and outperforms the rest by a large margin). Here, resampling helps the algorithms to perform better. The main reason is that our noise is very big and we need resampling to reduce the noise of our sampled, accepted, and not discarded points.

Next, we plot the reciprocals of the estimates of the bound (28) on the Pinter 10D and Rosenbrock 20D problems with high noise in Figure 10. The data in Figure 10 suggests that event E is not a rare event for the Pinter 10D and Rosenbrock 20D problems with high noise and hence that resampling is helpful in this case. This is consistent with our numerical results in Figures 8 and 9.

The numerical results presented in this section suggest that if we are equivocal between resampling versus non-resampling when applying our ASRD framework to solve optimization problems, one suggestion is to run the pre-processing procedure described above and base the decision on the results. For example, the bound (28) developed in this section is useful if and only if y is greater than 1 and suggests that E is rare if y is significantly larger than 1. Therefore, one possible criterion to choose between resampling versus non-resampling is: If $y > d$, where $d > 1$ is a chosen threshold value, for at least half the points plotted, then do not resample, otherwise resample. From our numerical results, we suggest $d \in [40, 70]$. Nevertheless, the optimal threshold value to use in choosing between resampling versus non-resampling

needs further investigation.

3.4 Conclusions

In this chapter, we propose and analyze a new random search algorithm, called Adaptive Search with Resampling and Discarding (ASRD), for continuous simulation optimization. The method is shown to converge to the optimal solution set almost surely with mild conditions on the algorithm parameters and the underlying problem; thus it can be applied to a wide variety of problems. Our approach improves upon the adaptive search with resampling (ASR) algorithm of Andradóttir and Prudius [13] in that (i) we develop a scheme to discard points that appear promising early on in the search but become inferior as the number of sampled points grows, and (ii) we use a time-varying criterion for accepting new, promising points, as opposed to the time-homogeneous acceptance criterion of Andradóttir and Prudius [13]. We provide numerical results showing that ASRD improves the convergence speed by a large margin compared to ASR. Another advantage of ASRD is that it requires much less memory than ASR (because we discard bad points in ASRD, whereas ASR will keep all accepted points). Finally, we show that whether it is desirable to conduct resampling or not depends on the problem, and we derive an indicator that compares the magnitude of the noise with the magnitude of the improvement in objective function values to determine whether resampling is desirable or not.

CHAPTER IV

SIMULATION-BASED CONTINUOUS OPTIMIZATION WITH STOCHASTIC CONSTRAINTS

4.1 Introduction

In this chapter, we present a framework for solving continuous simulation optimization problems with stochastic constraints, called Adaptive Search with Discarding and Penalization (ASDP). Optimization problems with stochastic constraints are very useful, as they allow for the consideration of multiple stochastic objectives. However, they are also challenging to solve due to the uncertainty involved in the constraints. Even checking feasibility of a given solution might be hard if the solution lies close to the boundary of the stochastic constraints. Sample average approximation can be used to estimate the feasibility of a solution. However, there is no convergence guarantee (especially from inside the feasible region) if only sample average approximation is used. Rather than estimating feasibility and optimizing the objective function separately, ASDP converts the objective function and stochastic constraints into a series of penalty functions, and consequently changes the original optimization problem into a sequence of stochastic optimization problems without stochastic constraints.

This chapter is organized as follows. In Section 4.2, we present our ASDP algorithm. In Section 4.3, we prove its almost sure convergence, discuss the needed assumptions, and propose an efficient acceptance criterion. In Section 4.4, we provide a numerical study. In Section 4.5, we summarize the main contributions of this chapter. An early version of this chapter can be found in [40].

4.2 The Algorithm

In this section, we present our ASDP algorithm for continuous simulation optimization problem with stochastic constraints. We start by introducing some notation. For all $\theta \in \Theta$ and $k \in \mathbb{N}$, let $N_k(\theta)$ be the number of observations of the objective function $f(\theta)$ as well as the constraint functions $g_j(\theta)$ by the end of iteration k, and let $S_k(\theta)$ be the sum of these $N_k(\theta)$ observations of $f(\theta)$ and $S_{j,k}(\theta)$ be the sum of these $N_k(\theta)$ observations of $g_j(\theta)$ for all $j \in \mathcal{C}$. Also, for all $\theta \in \Theta$, $j \in \mathcal{C}$, and $k \in \mathbb{N}$, let $\hat{f}_k(\theta) = S_k(\theta)/N_k(\theta)$ and $\hat{g}_{j,k}(\theta) = S_{j,k}(\theta)/N_k(\theta)$.

Since there are several existing simulation optimization algorithms to solve the unconstrained simulation optimization problem (1) (e.g., ASRD in Chapter 3, etc.), one natural question that arises here is, why not apply one of the existing frameworks to (2) and use estimated constraint function values to test feasibility (based on how the estimates compare with the bounds b_j for $j \in \mathcal{C}$)? In the following example, we will show that no matter how many constraint function observations one obtains during the simulation experiment, it is possible that the estimated optimal solution will be infeasible infinitely often.

Example 4.2.1. *Consider the optimization problem* (2) *with* $f(\theta) = \theta$ *and* $g(\theta) = \theta \leq 5$, *where* $\theta \in \Theta = [-10, 10]$, $h(\theta, X(\omega)) = f(\theta)$, $u(\theta, Y(\omega)) = g(\theta) + Y(\omega)$, *and* $Y(\omega)$ *is a* $\mathcal{N}(0, 100)$ *random variable. The optimal value of this constrained problem is* $f^* = 5$ *at* $\theta^* = 5$.

Let θ_i *be the candidate solution sampled in iteration* i, *and assume we obtain* $K(i) \geq 1$ *independent observations on* $g(\theta_i)$ *in iteration* i. *Choose the sampling strategy as follows: at iteration* i, *select point* $\theta_i = 5 + \frac{1}{\sqrt{K(i)}}$ *when* i *is even, and select point* $\theta_i = 5$ *(the optimal solution) when* i *is odd.*

At an even iteration i, $\hat{g}_i(\theta_i)$ *follows a* $\mathcal{N}(5 + \frac{1}{\sqrt{K(i)}}, \frac{100}{K(i)})$ *distribution, and hence* $P(\hat{g}_i(\theta_i) \leq 5) = P(\mathcal{N}(0,1) \leq -0.1) = \Phi(-0.1) > 0.4$. *Therefore, at any even step* i, *at least one infeasible point* (θ_i) *passes the feasibility test with probability* 0.4, *and*

the event is independent of the past if simulations are conducted independently in different iterations. Moreover, since there is no noise on the objective function, a point sampled in an even step will be selected as the estimate of the optimal solution once it passes the feasibility test. Since $\sum_{i=1}^{\infty} P(\hat{g}_{2i}(\theta_{2i}) \leq 5) = \infty$*, from the second Borel-Cantelli lemma, we know that the estimated optimal solution will almost surely be infeasible infinitely often.*

From the above Example 4.2.1, we can see that in order to obtain almost sure convergence from inside the feasible region, additional efforts should be made besides purely testing feasibility based on sample average approximation.

Let $\{V(i)\}_{i=1}^{\infty}$ be a strictly increasing sequence of positive integers with $V(1) = 1$. In our ASDP Algorithm, we alternate between adaptively sampling from the feasible region (if the current iteration number is equal to some element in the sequence $\{V(i)\}_{i=1}^{\infty}$), or resampling a previously sampled point (otherwise). After a new point has been sampled, we decide whether or not to accept the newly sampled point. The objective is to include sampled points that appear promising (i.e., appear feasible and the estimated objective function value is good), and reject the rest. Simultaneously, we ensure that we have collected enough objective function observations at each sampled point under consideration. Then we update the estimate of the optimal solution, and those points exhibiting inferior qualities are discarded. The reason we only update the estimate of the optimal solutions when $k = V(i)$, is because this allows us to guarantee almost sure convergence under weaker conditions than updating the estimate of the optimal solution at every iteration. More explanation can be found in Section 3.2.2.

To resolve the problem described in Example 4.2.1 due to the randomness involved in the constraints in problem (2), a penalty addressing the feasibility of estimated constraints is added to the estimate of the objective function as follows. Let $\{\lambda_i\}_{i=1}^{\infty}$, $\{\xi_i\}_{i=1}^{\infty}$ be two sequences of positive real numbers. For all $\theta \in \Theta$ and $i \in \mathbb{N}^+$, at

iteration $V(i)$, define

$$F_i(\theta) = \hat{f}_{V(i)}(\theta) - \lambda_i G_i(\theta, \xi_i),$$

where $G_i(\theta, \xi_i) = \mathbb{1}_{\{\sum_{j \in \mathcal{C}} \mathbb{1}_{\{\hat{g}_{j,V(i)}(\theta) > b_j - \xi_i\}} \geq 1\}}$ and $\mathbb{1}_A$ is 1 if event A is true, 0 otherwise. The motivation is: since some of the constraints cannot be evaluated exactly, we replace the constrained simulation optimization problem (2) by a single function consisting of the original estimate of the objective function $\hat{f}_{V(i)}(\theta)$, plus an additional term $\lambda_i G_i(\theta, \xi_i)$, which is positive when the current point θ either appears to be infeasible or shows ambiguity between feasible and infeasible (meaning that θ appears to be feasible but is very close to the boundary). Here we use the sequence $\{\xi_i\}_{i=1}^{\infty}$ to control our criteria to test feasibility, and $\{\lambda_i\}_{i=1}^{\infty}$ to control the scale of penalty when $G_i(\theta, \cdot)$ is positive. Returning to Example 4.2.1, the reason that infeasible points are selected as the estimate of the optimal solution is that no penalization is added to points that shows ambiguity between feasible and infeasible. The penalty term, $\lambda_i G_i(\theta, \xi_i)$, helps us solve this issue in that the points sampled in even iterations will be penalized (since they are very close to the boundary), and will be discarded eventually given appropriate conditions on algorithm parameters. We will describe the detailed algorithm below.

Next, let $\{K(i)\}_{i=1}^{\infty}$ be a nondecreasing sequence of positive integers. Let Θ_i be the set of solutions sampled and accepted by the end of iteration $V(i)$ without discarding already accepted points. Let Θ_i^* denote the set of solutions sampled, accepted, and not discarded by the end of iteration $V(i)$. Let Θ_i^+ be the set of solutions sampled and accepted by iteration $V(i)$, and not discarded prior to the discarding procedure in iteration $V(i)$. The pseudo-code of our ASDP approach is given in Algorithm 2.

Notice that as the number of iterations i grows, the set Θ_i of accepted sampled points keeps growing. Consequently, the computational effort of collecting additional function observations for each point in Θ_i keeps growing. To solve this issue, we use the sequences $\{\eta_i\}_{i=1}^{\infty}$ and $\{\delta_i\}_{i=1}^{\infty}$ to develop a scheme to discard points that appear

Algorithm 2 Adaptive Search with Discarding and Penalization (ASDP)

Select $c > 0$, $\{\lambda_i\}_{i=1}^{\infty}$, $\{\xi_i\}_{i=1}^{\infty}$, $\{\eta_i\}_{i=1}^{\infty}$, and $\{\delta_i\}_{i=1}^{\infty}$, four sequences of positive real numbers, $\{K(i)\}_{i=1}^{\infty}$, a nondecreasing sequence of positive integers with $K(i) = \Omega(i^c)$, a sampling strategy, a resampling strategy, and an acceptance criterion. Let $\Theta_0^* = \emptyset$, $i = 1$, and $k = 0$.
while Stopping criterion is not satisfied **do**
    Let $k = k + 1$
    if $k = V(i)$ **then**
        Sample a solution θ_i from Θ using the sampling strategy
        Based on the acceptance criterion, decide whether to include θ_i in the set Θ_i^+, so that $\Theta_i^+ \in \{\Theta_{i-1}^*, \Theta_{i-1}^* \cup \{\theta_i\}\}$, and update $N_k(\theta_i)$, $S_k(\theta_i)$, and $S_{j,k}(\theta_i)$ if needed
        For each $\theta \in \Theta_i^+$, if $N_k(\theta) < K(i)$, obtain $K(i) - N_k(\theta)$ additional observations of $f(\theta)$ and $g_j(\theta)$ ($j \in \mathcal{C}$), and update $N_k(\theta)$, $S_k(\theta)$, and $S_{j,k}(\theta)$ accordingly
        Let $\Theta_i^* = \Theta_i^+$
        Select an estimate of the current best solution $\theta_i^* \in \arg\max_{\theta \in \Theta_i^*} F_i(\theta)$
        if $\hat{g}_{j,k}(\theta_i^*) \leq b_j - \eta_i, \forall j \in \mathcal{C}$ **then**
            For each $\theta \in \Theta_i^*$, if $F_i(\theta_i^*) - F_i(\theta) > \delta_i$, remove θ from Θ_i^* and update $\Theta_i^* = \Theta_i^* \backslash \{\theta\}$
        end if
        Let $i = i + 1$
    else
        Sample a solution θ from Θ_{i-1}^* using the resampling strategy
        Obtain additional estimates of $f(\theta)$, $g_j(\theta)$ ($j \in \mathcal{C}$), and update $N_k(\theta)$, $S_k(\theta)$, and $S_{j,k}(\theta)$
    end if
end while
Return θ_{i-1}^* as an estimate of the optimal solution.

promising early on in the search but become inferior as the number of sampled points grows. In particular, we use the sequence $\{\eta_i\}_{i=1}^{\infty}$ to measure the feasibility of the current best estimate of the optimal solution at iteration $V(i)$. If θ_i^* appears to be feasible and not too close to the boundary ($\hat{g}_{j,k}(\theta_i^*) \leq b_j - \eta_i, \forall j \in \mathcal{C}$), we use it to discard points whose estimated function values are worse than the current best solution by at least δ_i ($F_i(\theta) < F_i(\theta_i^*) - \delta_i$), otherwise no. The main reason to test feasibility before executing the discarding procedure is to minimize the probability of using infeasible points to discard promising and seemingly feasible candidate points.

In terms of the choice of the sequences $\{\delta_i\}$ and $\{\eta_i\}$, a reasonable decision is to let both δ_i and η_i decrease as the number of sampling iterations i grows. The intuition is that the threshold is originally set to be large due to the noise generated by simulation in the early stages. However, as the number of iterations grows, the noise tends to disappear according to the strong law of large numbers, and the sample average of each point tends to be more accurately reflect the true objective function value.

4.3 Theory

In the previous section, we motivated and described the ASDP algorithm. In this section, we present the convergence results for our ASDP algorithm. In detail, Section 4.3.1 introduces some preliminaries, including assumptions that are used in the convergence results, Section 4.3.2 proves the algorithm converges to the optimal solution from inside the feasible region almost surely, Section 4.3.3 also proves the algorithm converges to the optimal solution with probability one but without feasibility guarantee, and Sections 4.3.4 and 4.3.5 discuss how the assumptions under which our method is guaranteed to converge can be satisfied in practice.

4.3.1 Preliminaries

In this section, we present our main convergence result for Algorithm 2. As in Chapter 3, *i.o.* stands for "infinitely often" and *a.a.* stands for "almost always." Let $|S|$ denote the cardinality of a set S, and $\bar{S}$ denote the complement of a set S (with respect to Θ). Let $\Theta_{\mathcal{F}} = \{\theta \in \Theta | g_j(\theta) \leq b_j, \forall j \in \mathcal{C}\}$ and $\bar{\Theta}_{\mathcal{F}} = \{\theta \in \Theta | g_j(\theta) > b_j, \exists j \in \mathcal{C}\}$. For each $\epsilon, \Delta \in \mathbb{R}$, define $\Theta_\epsilon = \{\theta \in \Theta | f(\theta) \geq f^* - \epsilon\}$ and $\bar{\Theta}_\epsilon = \{\theta \in \Theta | f(\theta) < f^* - \epsilon\}$. We also define $\Theta_{\mathcal{F},\Delta} = \{\theta \in \Theta | g_j(\theta) \leq b_j - \Delta, \forall j \in \mathcal{C}\}$, $\bar{\Theta}_{\mathcal{F},\Delta} = \{\theta \in \Theta | g_j(\theta) > b_j - \Delta, \exists j \in \mathcal{C}\}$, and let $\Theta_{\epsilon,\Delta} = \Theta_\epsilon \cap \Theta_{\mathcal{F},\Delta}$ and $\bar{\Theta}_{\epsilon,\Delta} = \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},\Delta}$. (Observe that $\Theta_{\mathcal{F}} = \Theta_{\mathcal{F},0}$ and $\bar{\Theta}_{\mathcal{F}} = \bar{\Theta}_{\mathcal{F},0}$.) For $n \in \mathbb{N}$ and $\theta \in \Theta$, define θ to be a "near-optimal" point with respect to ϵ if $\theta \in \Theta_\epsilon$. Let $f_n(\theta)$ be the estimate of $f(\theta)$ obtained from a sample average of n independent observations of $f(\theta)$, and for all $j \in \mathcal{C}$, let $g_{j,n}(\theta)$ be the estimate of $g_j(\theta)$ obtained from a sample average of n independent observations of $g_j(\theta)$. We also need the following assumptions.

Assumption 4.3.1. *For each $\theta \in \Theta$, we can generate independent and unbiased observations $\{h(\theta, X_k(\omega))\}$ of $f(\theta)$, and $\{u_j(\theta, Y_{j,k}(\omega))\}$ of $g_j(\theta)$ for each $j \in \mathcal{C}$. Moreover, there exist $l, w \in \mathbb{N}\backslash\{0,1\}$ and $R \in \mathbb{R}^+$ such that $E[(h(\theta, X_k(\omega)) - f(\theta))^{2l}] \leq R$ and $E[(u_j(\theta, Y_{j,k}(\omega)) - g_j(\theta))^{2w}] \leq R$ for $j \in \mathcal{C}$, $\theta \in \Theta$, and $k \in \mathbb{N}^+$.*

Assumption 4.3.2. *The random elements used for estimating the objective function and constraints values (e.g., in steps 6, 7, and 16 of ASDP) are independent of the random elements used in the execution of algorithmic decisions (e.g., in steps 5 and 15 of ASDP).*

Assumption 4.3.3. *For each $\epsilon > 0$, there exists $\Delta(\epsilon) > 0$ such that $P(\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}, i.o.) = 1$.*

Assumption 4.3.1 imposes the finiteness of moments for the random variables under consideration in this chapter (i.e., the observations of the objective and constraint functions). Note that this assumption is weaker than assuming the existence of

moment-generating functions in neighborhoods of zero, where the latter corresponds to $l, w = \infty$ in this assumption. Assumption 4.3.2 imposes restrictions on the random elements used by the algorithm. It is an assumption about implementation that can always be satisfied and allows for the use of common random numbers to estimate the objective function and constraints values at different solutions.

Assumption 4.3.3 imposes restrictions on the "shape" of the objective and constraint functions, as well as the sampling strategies we can use, namely that there exist "near-optimal" points with respect to ϵ in the interior of the feasible region, and we are able to find them (e.g., if we sample uniformly from a continuous feasible space, an objective function with an isolated optimal solution would violate Assumption 4.3.3). Moreover, since we discard points in multiple steps in our algorithms, we could possibly discard all "near-optimal" points with respect to ϵ if we do estimation poorly (see Example 3.2.1), in which case we need to be able to find "near-optimal" interior points with respect to ϵ again. Therefore we assume we can find "near-optimal" points with respect to ϵ interior to the feasible region infinitely often with probability one. We will show how Assumption 4.3.3 can be verified in Sections 4.3.4 and 4.3.5.

Before providing our convergence analysis for Algorithm 2, we need the following lemma:

Lemma 4.3.1. *Suppose the assumptions of Theorem 4.3.1 hold. Let* $C = \liminf_{i\to\infty}[\lambda_i - (\bar{f} - f^*)]/2$. *Given* $0 < \epsilon < C$, $0 < \epsilon' \leq \epsilon$, $\Delta \geq 0$, $\Delta' < \Delta$, *and* $\Delta + \epsilon - \epsilon' > 0$, *then, for each* $\theta, \theta' \in \Theta$, *and* $i \in \mathbb{N}^+$, *we have:*

$$P\left(\hat{f}_{V(i)}(\theta) - \hat{f}_{V(i)}(\theta') \geq \Delta, \theta \in \bar{\Theta}_\epsilon, \theta' \in \Theta_{\epsilon'}\right) \leq \frac{Const}{(\Delta + \epsilon - \epsilon')^{2l} i^{c(l-1)}}, \tag{29}$$

$$P\left(\hat{f}_{V(i)}(\theta) - \hat{f}_{V(i)}(\theta') \geq \lambda_i, \theta' \in \Theta_\epsilon\right) \leq \frac{Const}{i^{c(l-1)}}, \tag{30}$$

$$P\left(\theta \in \Theta_{\mathcal{F},\Delta}, G_i(\theta, \Delta') = 1\right) \leq \frac{Const}{(\Delta - \Delta')^{2w} i^{c(w-1)}}, \tag{31}$$

$$P\left(\theta \in \bar{\Theta}_{\mathcal{F},\Delta'}, G_i(\theta, \Delta) = 0\right) \leq \frac{Const}{(\Delta - \Delta')^{2w} i^{c(w-1)}}, \tag{32}$$

where "Const" denotes some constant positive number.

Proof. Consider

$$(29) \leq P\left(\hat{f}_{V(i)}(\theta) - \hat{f}_{V(i)}(\theta') \geq \Delta, f(\theta) < f^* - \epsilon, f(\theta') \geq f^* - \epsilon'\right)$$
$$\leq P\left(\hat{f}_{V(i)}(\theta) - f(\theta) + f(\theta') - \hat{f}_{V(i)}(\theta') > \Delta + \epsilon - \epsilon'\right)$$
$$\leq P\left(|\hat{f}_{V(i)}(\theta) - f(\theta)| > (\Delta + \epsilon - \epsilon')/2\right) + P\left(|f(\theta') - \hat{f}_{V(i)}(\theta')| > (\Delta + \epsilon - \epsilon')/2\right), \tag{33}$$

$$(30) \leq P\left(\hat{f}_{V(i)}(\theta) - \hat{f}_{V(i)}(\theta') \geq \lambda_i, f(\theta) \leq \bar{f}, f(\theta') \geq f^* - \epsilon\right)$$
$$\leq P\left(\hat{f}_{V(i)}(\theta) - f(\theta) + f(\theta') - \hat{f}_{V(i)}(\theta') \geq \lambda_i - (\bar{f} - f^*) - \epsilon\right), \tag{34}$$

$$(31) \leq P\left(\bigcup_{j\in\mathcal{C}}\{\hat{g}_{j,V(i)}(\theta) \geq b_j - \Delta', g_j(\theta) \leq b_j - \Delta\}\right)$$
$$\leq P\left(\bigcup_{j\in\mathcal{C}}\{\hat{g}_{j,V(i)}(\theta) - g_j(\theta) \geq \Delta - \Delta', \}\right)$$
$$\leq \sum_{j\in\mathcal{C}} P\left(|\hat{g}_{j,V(i)}(\theta) - g_j(\theta)| \geq \Delta - \Delta'\right), \tag{35}$$

$$(32) \leq P\left(\bigcup_{j\in\mathcal{C}}\{\hat{g}_{j,V(i)}(\theta) < b_j - \Delta, g_j(\theta) > b_j - \Delta'\}\right)$$
$$\leq \sum_{j\in\mathcal{C}} P\left(g_j(\theta) - \hat{g}_{j,V(i)}(\theta) > \Delta - \Delta'\right). \tag{36}$$

Since $\liminf_{i\to\infty} \lambda_i > \bar{f} - f^*$ and $0 < \epsilon < C$, we know that there exists $\bar{N} \in \mathbb{N}^+$ such that for $i \geq \bar{N}$, we have $\lambda_i > \bar{f} - f^*$. Using Assumption 4.3.1 and the same methodology as in deriving $(8) - (11)$ in Section 3.2.2, we know that

$$(29) \leq (33) \leq \frac{Const}{i^{c(l-1)}},$$

$$(30) \leq (34) \leq P\left(|\hat{f}_{V(i)}(\theta) - f(\theta) + f(\theta') - \hat{f}_{V(i)}(\theta')| \geq \epsilon\right)$$
$$\leq P\left(|\hat{f}_{V(i)}(\theta) - f(\theta)| \geq \epsilon/2\right) + P\left(|\hat{f}_{V(i)}(\theta') - f(\theta')| \geq \epsilon/2\right) \leq \frac{Const}{i^{c(l-1)}},$$

$$(31) \leq (35) \leq \frac{Const}{(\Delta - \Delta')^{2w} i^{c(w-1)}},$$

$$(32) \leq (36) \leq \sum_{j \in \mathcal{C}} P\left(|g_j(\theta) - \hat{g}_{j,V(i)}(\theta)| \geq \Delta - \Delta'\right) \leq \frac{Const}{(\Delta - \Delta')^{2w} i^{c(w-1)}}.$$

This completes the proof. □

4.3.2 Almost Sure Convergence from Inside the Feasible Region

Now we present our main theorem in this chapter.

Theorem 4.3.1. *Suppose Assumptions 4.3.1, 4.3.2, and 4.3.3 hold. Choose* $\{\lambda_i\}_{i=1}^{\infty}$ *such that* $\liminf_{i\to\infty} \lambda_i > \bar{f} - f^*$. *Let* $\delta_i = \Omega(i^{-\gamma_\delta})$, $\xi_i = \Omega(i^{-\gamma_\xi})$, *and* $\eta_i \geq \tau\xi_i$ *for each* $i \in \mathbb{N}^+$, *where* $\gamma_\xi > 0$ *and* $\tau > 1$. *If* $c(l-1) > 3$, $c(l-1) - 2\gamma_\delta l > 3$, *and* $c(w-1) - 2\gamma_\xi w > 3$, *then* $f(\theta_i^*) \to f^*$ *and* $\theta_i^* \in \Theta_{\mathcal{F}}$ *almost surely as* $i \to \infty$.

Theorem conditions $c(l-1) > 3$, $c(l-1) - 2\gamma_\delta l > 3$, and $c(w-1) - 2\gamma_\xi w > 3$ imply that given l (which characterizes the magnitude of the objective function noise) and w (which characterizes the magnitude of the constraint function noise), we need to choose c large enough ($c > \frac{3}{l-1}$), and choose γ_δ and γ_ξ appropriately ($\gamma_\delta < \frac{c(l-1)-3}{2l}$ and $\gamma_\xi < \frac{c(w-1)-3}{2w}$) to guarantee the convergence. Since $\gamma_\xi > 0$, $c(w-1) - 2\gamma_\xi w > 3$ implies $c(w-1) > 3$ must hold. Moreover, for larger l and w (i.e., less volatile noise), the condition on $\{\delta_i\}$ and $\{\xi_i\}$ are less restrictive. When the noise is well behaved, meaning $l, w = \infty$, we only need $c > 0$ and $\gamma_\delta, \gamma_\xi < c/2$.

In the following, we prove Theorem 4.3.1

Proof. Fix $0 < \epsilon < C$ (where C is defined in Lemma 4.3.1), and $1 < \kappa < \tau$. In order to prove the algorithm converges from inside $\Theta_{\mathcal{F}}$ almost surely, it suffices to show the following:

$$P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, i.o.\right) = 0.$$

We have:

$$P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, i.o.\right)$$
$$\leq P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, i.o.\right) + P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset, i.o.\right).$$

It suffices to show that (a) $P(\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, i.o.) = 0$ and (b) $P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset, i.o.) = 0$. Note that (a) ensures that all near-optimal interior points are not discarded infinitely often, and (b) ensures that the algorithm does a good job with estimation so that the estimate θ_i^* of the optimal solution is selected well when near-optimal interior points are available. In the following, we will repeatedly use Lemma 4.3.1 to bound the probability of the events under investigation.

We start by considering (a). By Assumption 4.3.3 and the fact that $\xi_i \to 0$ as $i \to \infty$, we have

$$\begin{aligned}
&P\left(\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, i.o.\right) \\
&= P\left(\{\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, i.o.\} \cap \{\theta_i \in \Theta_i \cap \Theta_{\epsilon/2,\Delta(\epsilon/2)}, i.o.\}\right) \\
&\leq P\left(\{\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, i.o.\} \cap \{\Theta_i^+ \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset, i.o.\}\right). \qquad (37)
\end{aligned}$$

It is not difficult to see that if we have near-optimal (with respect to $\epsilon/2$) interior (by $\kappa\xi_i$) points infinitely often prior to executing the discarding step (event $\{\Theta_i^+ \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset, i.o.\}$), but simultaneously we do not have such points in our sampled, accepted, and not discarded solution set infinitely often (event $\{\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, i.o.\}$), then it must happen infinitely often that there exist near-optimal points (with respect to $\epsilon/2$) interior (by $\kappa\xi_i$) in the set of sampled, accepted, and not discarded prior to the execution of discarding procedure, and we discard all such points (event $\{\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset, i.o.\}$). Hence we have:

$$(37) \leq P\left(\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset, i.o.\right). \qquad (38)$$

For each $i \in \mathbb{N}^+$, since $\eta_i = \tau\xi_i$, we consider

$$
\begin{aligned}
&P\left(\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset\right) \\
&\leq P\left(\bigcup_{\theta\in\Theta_i^+}\bigcup_{\theta'\in\Theta_i^+}\left\{F_i(\theta') - F_i(\theta) > \delta_i, \theta' \notin \Theta_{\epsilon/2,\kappa\xi_i}, G_i(\theta',\tau\xi_i) = 0, \theta \in \Theta_{\epsilon/2,\kappa\xi_i}\right\}\right) \\
&= P\left(\bigcup_{\theta\in\Theta_i^+}\bigcup_{\theta'\in\Theta_i^+}\left\{F_i(\theta') - F_i(\theta) > \delta_i, \theta' \in \bar{\Theta}_{\epsilon/2} \cup \bar{\Theta}_{\mathcal{F},\kappa\xi_i}, G_i(\theta',\tau\xi_i) = 0, \theta \in \Theta_{\epsilon/2,\kappa\xi_i}\right\}\right).
\end{aligned} \tag{39}
$$

For each $i \in \mathbb{N}^+$, let $\tilde{\Theta}_i$ be the set of sampled points by the end of iteration $V(i)$; note that $|\tilde{\Theta}_i| \leq i$ (in general $|\tilde{\Theta}| = i$ would be expected, unless the sampling strategy allows for resampling). As in Andradóttir and Prudius [13], suppose that if a sampled point is rejected or discarded, we still collect additional observations at this point to ensure that it has enough observations collected at it (i.e., by the end of iteration $V(i)$ it has at least $K(i)$ observations). Although we collect additional observations at the points in $\tilde{\Theta}_i \backslash \Theta_i^+$, we do not use them for making decisions concerning the evolution of the algorithm. Thus collecting additional data at these points does not impact convergence, and in practice we would not collect this data.

Since $\Theta_i^+ \subseteq \tilde{\Theta}_i$, we have

$$
\begin{aligned}
(39) &\leq P\left(\bigcup_{\theta'\in\tilde{\Theta}_i}\bigcup_{\theta\in\tilde{\Theta}_i}\left\{F_i(\theta') - F_i(\theta) > \delta_i, \theta' \in \bar{\Theta}_{\epsilon/2} \cup \bar{\Theta}_{\mathcal{F},\kappa\xi_i}, G_i(\theta',\tau\xi_i) = 0, \theta \in \Theta_{\epsilon/2,\kappa\xi_i}\right\}\right) \\
&\leq \sum_{m=1}^{i}\sum_{n=1}^{i} P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2} \cup \bar{\Theta}_{\mathcal{F},\kappa\xi_i}, G_i(\theta_m,\tau\xi_i) = 0, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq \sum_{m=1}^{i}\sum_{n=1}^{i} P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, G_i(\theta_m,\tau\xi_i) = 0, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \qquad (40) \\
&\quad + \sum_{m=1}^{i}\sum_{n=1}^{i} P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\mathcal{F},\kappa\xi_i}, G_i(\theta_m,\tau\xi_i) = 0, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right). \qquad (41)
\end{aligned}
$$

From the definition of $G_i(\cdot,\cdot)$, for each $i \in \mathbb{N}^+$, $\theta \in \Theta$, and $\xi \geq 0$, we have that $G_i(\theta,\tau\xi) = 0$ implies that $G_i(\theta,\xi) = 0$ since $\tau > 1$. To obtain an upper bound on

(40), for each $m, n \in \mathbb{N}^+$ with $1 \leq m, n \leq i$, we consider

$$
\begin{aligned}
&P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, G_i(\theta_m, \tau\xi_i) = 0, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) > \delta_i - \lambda_i G_i(\theta_n, \xi_i), \theta_m \in \bar{\Theta}_{\epsilon/2}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&= P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}, G_i(\theta_n, \xi_i) = 0\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) > \delta_i - \lambda_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}, G_i(\theta_n, \xi_i) = 1\right) \\
&\leq P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, \theta_n \in \Theta_{\epsilon/2}\right) + P\left(\theta_n \in \Theta_{\mathcal{F},\kappa\xi_i}, G_i(\theta_n, \xi_i) = 1\right) \\
&\leq \frac{Const}{\delta_i^{2l} i^{c(l-1)}} + \frac{Const}{\xi_i^{2w} i^{c(w-1)}}, \qquad (42)
\end{aligned}
$$

where (42) follows from (29) and (31). To obtain an upper bound on (41), for each $m, n \in \mathbb{N}^+$ with $1 \leq m, n \leq i$, we consider

$$
\begin{aligned}
&P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\mathcal{F},\kappa\xi_i}, G_i(\theta_m, \tau\xi_i) = 0, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq P\left(\theta_m \in \bar{\Theta}_{\mathcal{F},\kappa\xi_i}, G_i(\theta_m, \tau\xi_i) = 0\right) \\
&\leq \frac{Const}{\xi_i^{2w} i^{c(w-1)}}, \qquad (43)
\end{aligned}
$$

where (43) follows from (32).

Next, consider (b). For each $i \in \mathbb{N}^+$, we have

$$
\begin{aligned}
&P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset) \\
&\leq P\left(\bigcup_{\theta' \in \Theta_i^*} \bigcup_{\theta \in \Theta_i^*} \left\{F_i(\theta') \geq F_i(\theta), \theta' \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \theta \in \Theta_{\epsilon/2,\kappa\xi_i}\right\}\right) \\
&= P\left(\bigcup_{\theta' \in \Theta_i^*} \bigcup_{\theta \in \Theta_i^*} \left\{\hat{f}_{V(i)}(\theta') - \hat{f}_{V(i)}(\theta) \geq \lambda_i[G_i(\theta', \xi_i) - G_i(\theta, \xi_i)], \theta' \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \theta \in \Theta_{\epsilon/2,\kappa\xi_i}\right\}\right) \\
&\leq P\left(\bigcup_{\theta' \in \tilde{\Theta}_i} \bigcup_{\theta \in \tilde{\Theta}_i} \left\{\hat{f}_{V(i)}(\theta') - \hat{f}_{V(i)}(\theta) \geq \lambda_i[G_i(\theta', \xi_i) - G_i(\theta, \xi_i)], \theta' \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \theta \in \Theta_{\epsilon/2,\kappa\xi_i}\right\}\right) \\
&\leq \sum_{m=1}^{i} \sum_{n=1}^{i} P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, \xi_i) - G_i(\theta_n, \xi_i)], \theta_m \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq \sum_{m=1}^{i} \sum_{n=1}^{i} P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, \xi_i) - G_i(\theta_n, \xi_i)], \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \qquad (44) \\
&\quad + \sum_{m=1}^{i} \sum_{n=1}^{i} P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, \xi_i) - G_i(\theta_n, \xi_i)], \theta_m \in \bar{\Theta}_{\mathcal{F}}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right). \qquad (45)
\end{aligned}
$$

To bound (44), for each $m, n \in \mathbb{N}^+$ with $1 \leq m, n \leq i$, we consider

$$
\begin{aligned}
&P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, \xi_i) - G_i(\theta_n, \xi_i)], \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq 0, G_i(\theta_m, \xi_i) = 0, G_i(\theta_n, \xi_i) = 0, \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq 0, G_i(\theta_m, \xi_i) = 1, G_i(\theta_n, \xi_i) = 1, \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta) \geq \lambda_i, G_i(\theta_m, \xi_i) = 1, G_i(\theta_n, \xi_i) = 0, \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq -\lambda_i, G_i(\theta_m, \xi_i) = 0, G_i(\theta_n, \xi_i) = 1, \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq 3P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq 0, \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2}\right) + P\left(\theta_n \in \Theta_{\mathcal{F},\kappa\xi_i}, G_i(\theta_n, \xi_i) = 1\right) \\
&\leq \frac{Const}{i^{c(l-1)}} + \frac{Const}{\xi_i^{2w} i^{c(w-1)}}, \qquad (46)
\end{aligned}
$$

where (46) follows from (29) and (31). To bound (45), for each $m, n \in \mathbb{N}^+$ with

$1 \leq m, n \leq i$, we consider

$$
\begin{aligned}
&P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, \xi_i) - G_i(\theta_n, \xi_i)], \theta_m \in \bar{\Theta}_{\mathcal{F}}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq 0, G_i(\theta_m, \xi_i) = 0, G_i(\theta_n, \xi_i) = 0, \theta_m \in \bar{\Theta}_{\mathcal{F}}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq 0, G_i(\theta_m, \xi_i) = 1, G_i(\theta_n, \xi_i) = 1, \theta_m \in \bar{\Theta}_{\mathcal{F}}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i, G_i(\theta_m, \xi_i) = 1, G_i(\theta_n, \xi_i) = 0, \theta_m \in \bar{\Theta}_{\mathcal{F}}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq -\lambda_i, G_i(\theta_m, \xi_i) = 0, G_i(\theta_n, \xi_i) = 1, \theta_m \in \bar{\Theta}_{\mathcal{F}}, \theta_n \in \Theta_{\epsilon/2,\kappa\xi_i}\right) \\
&\leq P\left(\theta_m \in \bar{\Theta}_{\mathcal{F}}, G_i(\theta_m, \xi_i) = 0\right) + 2P\left(\theta_n \in \Theta_{\mathcal{F},\kappa\xi_i}, G_i(\theta_n, \xi_i) = 1\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i, \theta_n \in \Theta_{\epsilon/2}\right) \\
&\leq \frac{Const}{\xi_i^{2w} i^{c(w-1)}} + \frac{Const}{i^{c(l-1)}}, \qquad (47)
\end{aligned}
$$

where (47) follows from (30), (31), and (32).

To sum up, we can see that (39) − (43) imply:

$$P(\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset) \leq \frac{Const}{\delta_i^{2l} i^{c(l-1)-2}} + \frac{Const}{\xi_i^{2w} i^{c(w-1)-2}},$$

and (44) − (47) imply:

$$P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset) \leq \frac{Const}{i^{c(l-1)-2}} + \frac{Const}{\xi_i^{2w} i^{c(w-1)-2}}.$$

Since $\delta_i = \Omega(i^{-\gamma_\delta})$, $\xi_i = \Omega(i^{-\gamma_\xi})$, where $\gamma_\xi > 0$, $c(l-1) > 3$, $c(l-1) - 2\gamma_\delta l > 3$, and $c(w-1) - 2\gamma_\xi w > 3$, it follows that $\sum_{i=1}^{\infty} P(\Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset) < \infty$ and $\sum_{i=1}^{\infty} P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F}}, \Theta_i^* \cap \Theta_{\epsilon/2,\kappa\xi_i} \neq \emptyset) < \infty$. It now follows from (37), (38), and the first Borel-Cantelli lemma that (a) and (b) are true. This completes the proof. □

Remark 4.3.1. *In Algorithm* 2*, we only sample one point in each sampling step (Step* 5*). However, after careful examination of the convergence proof of the algorithm, we notice that if we sample a fixed number of points in Step* 5*, the convergence result will not be affected.*

4.3.3 Almost Sure Convergence without Feasibility Guarantee

In this section, we provide a convergence result for Algorithm 2 under the condition $\xi_i = 0$ for all $i \in \mathbb{N}^+$.

From both Example 4.2.1 and Theorem 4.3.1, we know that $\xi_i > 0$ is necessary for convergence from inside the feasible region. If we would like to explore convergence results under $\xi_i = 0$, we need to allow for convergence from outside the feasible region. However, in Example 4.3.1, we will show that even if convergence from outside the feasible region is allowed and Assumptions 4.3.1, 4.3.2, and 4.3.3 hold, convergence to the optimal need not occur.

Example 4.3.1. *Consider the optimization problem (2) with*

$$f(\theta) = \begin{cases} \theta + 10 & \text{if } -10 \leq \theta < -5, \\ 10 & \text{if } -5 \leq \theta \leq 0, \\ -\theta + 10 & \text{if } 0 \leq \theta \leq 5, \\ \theta + 15 & \text{if } 5 < \theta \leq 10, \end{cases}$$

and

$$g(\theta) = \theta \leq 5,$$

where $\theta \in \Theta = [-10, 10]$. *Then* $\Theta_{\mathcal{F}} = [-10, 5]$, *and we assume that* $h(\theta, X(\omega)) = f(\theta)$, $u(\theta, Y(\omega)) = g(\theta) + Y(\omega)$, *where* $Y(\omega)$ *is a* $\mathcal{N}(0, 100)$ *random variable. The optimal value of this constrained problem is* $f^* = 10$ *at* $\theta^* \in [-5, 0]$. *Let* θ_i *be the candidate solution sampled in iteration* i, *and assume we obtain* $K(i) \geq 1$ *independent observations on* $g(\theta_i)$ *in iteration* i. *At iteration* i, *select point* $\theta_i = 5 + \frac{1}{\sqrt{K(i)}}$ *when* i *is even, and select point* $\theta_i = -5$ *(an optimal solution) when* i *is odd (so that Assumption 4.3.3 is satisfied). Apply the ASDP algorithm with* $\xi_i = 0$ *for* $i \in \mathbb{N}^+$, *which means no penalty is added for points that pass the feasibility test. At an even iteration* i, $\hat{g}_i(\theta_i)$ *follows a* $\mathcal{N}(5 + \frac{1}{\sqrt{K(i)}}, \frac{100}{K(i)})$ *distribution,* $P(\hat{g}_i(\theta_i) \leq 5) = P(\mathcal{N}(0, 1) \leq -0.1) = \Phi(-0.1) > 0.4$, *and* $f(\theta_i) = 20 + \frac{1}{\sqrt{K(i)}}$. *Hence, at any even step* i, *at least one*

infeasible point (θ_i) passes the feasibility test with probability no smaller than 0.4*, and the event is independent of the past if simulations are conducted independently in different iterations. Moreover, since there is no noise in the objective function, a point sampled in an even step will be selected as the estimate of the optimal solution once it passes the feasibility test. Clearly,* $\lim_{i\to\infty} f(\theta_{2i}) = 20 > f^*$*,* $\lim_{i\to\infty} \theta_{2i} = 5$*, and* $f(5) = 5 < f^*$*. From the second Borel-Cantelli lemma, there is almost surely a subsequence* $\{i_k\}$ *such that* $\theta^*_{i_k} \to 5$ *as* $k \to \infty$*, which is far away from the true optimal region* $[-5, 0]$*.*

To rule out the situation described in Example 4.3.1, we need the following assumption.

Assumption 4.3.4. *For each* $\epsilon > 0$*, there exists* $\Gamma(\epsilon) > 0$ *such that* $\Theta_{\mathcal{F},-\Gamma(\epsilon)} \subset \bar{\Theta}_{-\epsilon}$*.*

Assumption 4.3.4 assumes that there are no infeasible points that are both almost feasible and also significantly better than optimal.

Now we present our convergence result.

Theorem 4.3.2. *Suppose Assumptions* 4.3.1 − 4.3.4 *hold. Choose* $\{\lambda_i\}_{i=1}^{\infty}$ *such that* $\liminf_{i\to\infty} \lambda_i > \bar{f} - f^*$*. Let* $\delta_i = \Omega(i^{-\gamma_\delta})$*,* $\eta_i = \Omega(i^{-\gamma_\eta})$*, where* $\gamma_\eta > 0$*, and* $\xi_i = 0$ *for* $i \in \mathbb{N}^+$*. If* $c(l-1) > 3$*,* $c(l-1) - 2\gamma_\delta l > 3$*, and* $c(w-1) - 2\gamma_\eta w > 3$*, then* $f(\theta^*_i) \to f^*$ *almost surely as* $i \to \infty$*.*

Proof. To prove $f(\theta^*_i) \to f^*$ almost surely as $i \to \infty$, it is sufficient to show that for any $\epsilon > 0$, we have:

$$P\left(\theta^*_i \in \bar{\Theta}_\epsilon \cup \Theta_{-\epsilon}, i.o.\right) = 0. \tag{48}$$

From Assumption 4.3.4, we know there exists $\Gamma(\epsilon) > 0$ such that $\Theta_{\mathcal{F},-\Gamma(\epsilon)} \subset \bar{\Theta}_{-\epsilon}$, and hence $\Theta_{-\epsilon} \subset \Theta_{\mathcal{F},-\Gamma(\epsilon)}$. It follows that

$$P\left(\theta^*_i \in \bar{\Theta}_\epsilon \cup \Theta_{-\epsilon}, i.o.\right) \le P\left(\theta^*_i \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, i.o.\right).$$

Therefore, it is sufficient to show that $P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, i.o.\right) = 0$.

We have:

$$
\begin{aligned}
&P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, i.o.\right) \\
&\leq P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, i.o.\right) + P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset, i.o.\right) \\
&\leq P\left(\Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, i.o.\right) + P\left(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset, i.o.\right).
\end{aligned}
$$

Hence, we need to show that (a) $P(\Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, i.o.) = 0$ and (b) $P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset, i.o.) = 0$.

We first consider (a). Using the same argument as for deriving (38) and the fact that $\eta_i \to 0$ as $i \to \infty$, we have:

$$
P\left(\Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, i.o.\right) \leq P\left(\Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset, i.o.\right). \tag{49}
$$

For each $i \in \mathbb{N}^+$, using the same method as in deriving (39) $-$ (43), and the fact that $\xi_i = 0$, we have

$$
\begin{aligned}
&P\left(\Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset\right) \\
&\leq \sum_{m=1}^{i}\sum_{n=1}^{i} P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, G_i(\theta_m,\eta_i) = 0, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right) \qquad (50) \\
&\quad + \sum_{m=1}^{i}\sum_{n=1}^{i} P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\mathcal{F},\eta_i/2}, G_i(\theta_m,\eta_i) = 0, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right). \qquad (51)
\end{aligned}
$$

Moreover, for each $m, n \in \mathbb{N}^+$ with $1 \leq m, n \leq i$,

$$
\begin{aligned}
&P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, G_i(\theta_m,\eta_i) = 0, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right) \\
&\leq P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \delta_i, \theta_m \in \bar{\Theta}_{\epsilon/2}, \theta_n \in \Theta_{\epsilon/2}\right) + P\left(\theta_n \in \Theta_{\mathcal{F},\eta_i/2}, G_i(\theta_n, 0) = 1\right) \\
&\leq \frac{Const}{\delta_i^{2l} i^{c(l-1)}} + \frac{Const}{\eta_i^{2w} i^{c(w-1)}}, \qquad (52)
\end{aligned}
$$

where (52) follows from (29) and (31). Similarly, for each $m, n \in \mathbb{N}^+$ with $1 \leq m, n \leq$

i,

$$
\begin{aligned}
&P\left(F_i(\theta_m) - F_i(\theta_n) > \delta_i, \theta_m \in \bar{\Theta}_{\mathcal{F},\eta_i/2}, G_i(\theta_m, \eta_i) = 0, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right) \\
&\leq P\left(\theta_m \in \bar{\Theta}_{\mathcal{F},\eta_i/2}, G_i(\theta_m, \eta_i) = 0\right) \\
&\leq \frac{Const}{\eta_i^{2w} i^{c(w-1)}}, \qquad (53)
\end{aligned}
$$

where (53) follows from (32).

Now, consider (b). For each $i \in \mathbb{N}^+$, using the same approach as in deriving $(44) - (47)$ and $\xi_i = 0$, we have

$$
\begin{aligned}
&P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset) \\
&\leq P\left(\bigcup_{\theta' \in \Theta_i^*} \bigcup_{\theta \in \Theta_i^*} \left\{F_i(\theta') \geq F_i(\theta), \theta' \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \theta \in \Theta_{\epsilon/2,\eta_i/2}\right\}\right) \\
&\leq \sum_{m=1}^{i} \sum_{n=1}^{i} P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, 0) - G_i(\theta_n, 0)], \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right) \qquad (54) \\
&\quad + \sum_{m=1}^{i} \sum_{n=1}^{i} P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, 0) - G_i(\theta_n, 0)], \theta_m \in \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right). \qquad (55)
\end{aligned}
$$

We also have, for each $m, n \in \mathbb{N}^+$ with $1 \leq m, n \leq i$,

$$
\begin{aligned}
&P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, 0) - G_i(\theta_n, 0)], \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right) \\
&\leq 3P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq 0, \theta_m \in \bar{\Theta}_\epsilon, \theta_n \in \Theta_{\epsilon/2}\right) + P\left(\theta_n \in \Theta_{\mathcal{F},\eta_i/2}, G_i(\theta_n, 0) = 1\right) \\
&\leq \frac{Const}{i^{c(l-1)}} + \frac{Const}{\eta_i^{2w} i^{c(w-1)}}, \qquad (56)
\end{aligned}
$$

where (56) follows from (29) and (31). Similarly, for each $m, n \in \mathbb{N}^+$ with $1 \leq m, n \leq i$,

$$
\begin{aligned}
&P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i[G_i(\theta_m, 0) - G_i(\theta_n, 0)], \theta_m \in \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \theta_n \in \Theta_{\epsilon/2,\eta_i/2}\right) \\
&\leq P\left(\theta_m \in \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, G_i(\theta_m, 0) = 0\right) + 2P\left(\theta_n \in \Theta_{\mathcal{F},\eta_i/2}, G_i(\theta_n, 0) = 1\right) \\
&\quad + P\left(\hat{f}_{V(i)}(\theta_m) - \hat{f}_{V(i)}(\theta_n) \geq \lambda_i, \theta_n \in \Theta_{\epsilon/2}\right) \\
&\leq \frac{Const}{i^{c(w-1)}} + \frac{Const}{\eta_i^{2w} i^{c(w-1)}} + \frac{Const}{i^{c(l-1)}}, \qquad (57)
\end{aligned}
$$

where (57) follows from (30), (31), and (32).

From the above analysis, we can see that $(50)-(53)$ imply:

$$P(\Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset) \leq \frac{Const}{\delta_i^{2l} i^{c(l-1)-2}} + \frac{Const}{\eta_i^{2w} i^{c(w-1)-2}},$$

and $(54)-(57)$ imply:

$$P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset) \leq \frac{Const}{i^{c(l-1)-2}} + \frac{Const}{i^{c(w-1)-2}} + \frac{Const}{\eta_i^{2w} i^{c(w-1)-2}}.$$

Since $\delta_i = \Omega(i^{-\gamma_\delta})$, $\eta_i = \Omega(i^{-\gamma_\eta})$, where $\gamma_\eta > 0$, $c(l-1) > 3$, $c(l-1) - 2\gamma_\delta l > 3$, and $c(w-1) - 2\gamma_\eta w > 3$, it follows that $\sum_{i=1}^{\infty} P(\Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} = \emptyset, \Theta_i^+ \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset) < \infty$ and $\sum_{i=1}^{\infty} P(\theta_i^* \in \bar{\Theta}_\epsilon \cup \bar{\Theta}_{\mathcal{F},-\Gamma(\epsilon)}, \Theta_i^* \cap \Theta_{\epsilon/2,\eta_i/2} \neq \emptyset) < \infty$. It follows from (49), and the first Borel-Cantelli lemma that (a) and (b) are true. This completes the proof. □

4.3.4 Adaptive Random Search

From the analysis of Algorithm 2 in Section 4.3.1, we know that one key assumption to guarantee the convergence of our algorithm is Assumption 4.3.3; explicitly, given any $\epsilon > 0$, our sampling strategy should be able to repeatedly find "near optimal" points with respect to ϵ in the interior of the feasible region. In this section, we propose several random search strategies to satisfy Assumption 4.3.3.

Let G denote a distribution on the feasible region Θ. We need the following assumption.

Assumption 4.3.5. *For each $\epsilon > 0$, we have $G(\Theta_{\epsilon,\Delta(\epsilon)}) > 0$.*

Assumption 4.3.5 imposes proper structure on the objective function, constraints, and the sampling distribution. We use the following example to illustrate the assumption.

Example 4.3.2. *Let $f(\theta) = -\theta^2$, $\Theta = [-1, 1]$, and further assume that the only constraint is $g(\theta) = \theta \leq 0$. The optimal solution is $\theta^* = 0$ with $f^* = 0$. For each $\epsilon > 0$, choose $\Delta(\epsilon) = \sqrt{\epsilon}/3$, then we have $\Theta_{\epsilon,\Delta(\epsilon)} = [-\sqrt{\epsilon}, -\sqrt{\epsilon}/3]$. If we let G be*

a uniform distribution on $[-1, 1]$*, then we have* $G(\Theta_{\epsilon,\Delta(\epsilon)}) = \sqrt{\epsilon}/3 > 0$*. Assumption* 4.3.5 *holds.*

Consider two sampling strategies, namely "random search" (at each sampling iteration, sample a point from the whole region Θ according to distribution G independent of everything) and "adaptive search" (at each sampling iteration, with probability $0 < p \leq 1$, sample a point using distribution G independent of everything, and with probability $1 - p$, sample a point using some local distribution which might be based on the currently available information). Under Assumption 4.3.5, we can show that if either random search or adaptive search is used and we accept every sampled point, then Assumption 4.3.3 holds. The formal results are as follows:

Proposition 4.3.1. *Under Assumption 4.3.5, if sampling strategy "random search" is used and we accept every sampled point, then Assumption 4.3.3 holds.*

Proof. Let $\epsilon > 0$. In each sampling step i, the probability of sampling a good point is $G(\Theta_{\epsilon,\Delta(\epsilon)})$, regardless of the past information. As we accept every sampled point, we have that $P(\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}) = G(\Theta_{\epsilon,\Delta(\epsilon)}) > 0$ from Assumption 4.3.5. Therefore: $\sum_{i=1}^{\infty} P(\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}) = \infty$. From the second Borel-Cantelli lemma, we know Assumption 4.3.3 holds. This completes the proof. □

Proposition 4.3.2. *Under Assumption 4.3.5, if sampling strategy "adaptive search" is used and we accept every sampled point, then Assumption 4.3.3 holds.*

Proof. Let $\epsilon > 0$. Since we accept every point we sample, we have that $P(\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}) \geq pG(\Theta_{\epsilon,\Delta(\epsilon)}) > 0$ from Assumption 4.3.5 and $p > 0$. Therefore, $\sum_{i=1}^{\infty} P(\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}) = \infty$. Again, from the second Borel-Cantelli lemma, we know Assumption 4.3.3 holds. This completes the proof. □

4.3.5 Acceptance Criterion

In this section, we describe an acceptance criterion for our ASDP algorithm. A good acceptance criterion can efficiently allow the algorithm to avoid spending much effort on inferior solutions or infeasible solutions or both. Andradóttir and Prudius [13] proposed an acceptance criterion based on a fixed number of objective function observations collected at the newly sampled solution for their ASR framework (see Section 3.2.3). On the other hand, we proposed an acceptance criterion based on a time-varying number of objective function observations collected at the newly sampled solution for our ASRD framework in Section 3.2.3. The basic idea of both acceptance criteria is that if an objective function estimate (based on the chosen number of observations) at a newly sampled point is at least as good as the estimated objective function value at the best solution found so far minus a positive indifference parameter, we accept this point, otherwise reject it immediately.

Here, since we are considering the constrained simulation optimization problem (2), we not only need to consider the performance of newly sampled points, but also need to take into account the feasibility of newly sampled points. Motivated by the idea of Andradóttir and Prudius [13] and the acceptance criterion in Section 3.2.3, we suggest the following acceptance criterion: Let $\alpha > 0$, $\beta > 0$, and $\{H(i)\}$ be a sequence of positive integers. We obtain $H(i)$ independent observations of $f(\theta_i)$ and $g_j(\theta)$ ($j \in \mathcal{C}$), after we sample new point θ_i. We always accept the first sampled point, even if it appears infeasible (we need a starting point of our algorithm and we can add penalty to the first point later on). For $i \geq 2$, the newly sampled solution θ_i is included in the set Θ_i^+ of sampled, accepted, and not discarded points in iteration $V(i)$ if it passes two stages:

$$\begin{cases} \text{Stage One: } g_{j,H(i)}(\theta_i) \leq b_j + \beta, \text{ for } \forall j \in \mathcal{C}, \\ \text{Stage Two: } f_{H(i)}(\theta_i) \geq \hat{f}_{V(i-1)}(\theta^*_{i-1}) - \alpha, \text{ if } \hat{g}_{j,V(i-1)}(\theta^*_{i-1}) \leq b_j - \eta_{i-1}, \forall j \in \mathcal{C}. \end{cases}$$

Otherwise, reject this point. Notice that once θ_i passes the "feasibility test" (stage

one), we accept it if the objective function estimate based on our observations, $f_{H(i)}(\theta_i)$, is at least as good as the estimated objective function value at the best solution found so far, which is $\hat{f}_{V(i-1)}(\theta^*_{i-1})$, minus a positive indifference parameter if our current best solution appears to be feasible ($\hat{g}_{j,V(i-1)}(\theta^*_{i-1}) \leq b_j - \eta_{i-1}, \forall j \in \mathcal{C}$), otherwise we reject it immediately. The motivation is, at iteration $i > 1$, when a new point θ_i is sampled, we first need to consider whether it is feasible or not. If there is not strong evidence that θ_i is infeasible (i.e., θ_i passes the Stage One test), then we decide whether this point is promising or not through the Stage Two test. The reason we test feasibility of θ^*_{i-1} before using it to reject any potential candidate solutions is to lower the risk of using near-boundary infeasible solutions to reject promising feasible new points. We next verify the validity of the acceptance criterion.

Proposition 4.3.3. *Suppose Assumptions 4.3.1 and 4.3.2 hold,* $0 < \epsilon < \alpha$, *let* $\eta_i = \Omega(i^{-\gamma_\eta})$. *Select* c *and* γ_η *to satisfy* $c(l-1) > 2$ *and* $c(w-1) - 2w\gamma_\eta > 2$, *and choose* $\{\lambda_i\}_{i=1}^{\infty}$ *such that* $\liminf_{i\to\infty} \lambda_i > \bar{f} - f^*$. *We also sample points in* $\Theta_{\epsilon,\Delta(\epsilon)}$ *infinitely often with probability one. If our acceptance criterion is used and* $H(i) = \Omega(i^q)$ *for all* i, *where* q *is a positive real number satisfying* $ql > 2$ *and* $qw > 1$, *then Assumption 4.3.3 holds.*

Proof. Our goal is to show that $\exists \Delta(\epsilon) > 0$ s.t. $P(\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}, i.o.) = 1$, or equivalently $P(\{\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\}^c) = 0$. Since we sample points in $\Theta_{\epsilon,\Delta(\epsilon)}$ infinitely often with probability one, we have $P(\theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, i.o.) = 1$. Therefore:

$$\begin{aligned} P\left(\{\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\}^c\right) &= P\left(\{\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\}^c \bigcap \{\theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\}\right) \\ &= P\left(\{\theta_i \notin \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}, a.a.\} \bigcap \{\theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\}\right) \\ &\leq P\left(\theta_i \notin \Theta_i, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\right). \end{aligned} \tag{58}$$

To prove that (58) equals zero, let $i > 1$ (we do not consider $i = 1$ is because we

always accept the first sampled point). Consider

$$
\begin{aligned}
&P\left(\theta_i \notin \Theta_i, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}\right) \\
&= P\left(\left[\left\{\bigcup_{j\in\mathcal{C}}\{g_{j,H(i)}(\theta_i) > b_j + \beta\}\right\} \bigcup \left\{\bigcup_{\theta\in\Theta^*_{i-1}} \left\{\hat{f}_{V(i-1)}(\theta) - f_{H(i)}(\theta_i) > \alpha, G_{i-1}(\theta,\eta_{i-1}) = 0\right\}\right\}\right] \bigcap \{\theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}\}\right) \\
&\leq P\left(\bigcup_{\theta\in\Theta^*_{i-1}} \left\{\hat{f}_{V(i-1)}(\theta) - f_{H(i)}(\theta_i) > \alpha, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, G_{i-1}(\theta,\eta_{i-1}) = 0\right\}\right) \\
&\quad + P\left(\bigcup_{j\in\mathcal{C}} \{g_{j,H(i)}(\theta_i) > b_j + \beta, \theta_i \in \Theta_{\mathcal{F},\Delta(\epsilon)}\}\right) \\
&\leq \sum_{m=1}^{i-1} P\left(\hat{f}_{V(i-1)}(\theta_m) - f_{H(i)}(\theta_i) > \alpha, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, G_{i-1}(\theta_m,\eta_{i-1}) = 0\right) \qquad (59) \\
&\quad + P\left(\bigcup_{j\in\mathcal{C}} \{g_{j,H(i)}(\theta_i) - g(\theta_i) > \beta\}\right). \qquad (60)
\end{aligned}
$$

Since $\liminf_{i\to\infty} \lambda_i > \bar{f} - f^*$, there exists $N^* > 1$, such that for $i > N^*$, $\lambda_{i-1} > \bar{f} - f^*$. Next consider $i > N^*$. For each $m = 1, \ldots, i-1$, we have

$$
\begin{aligned}
&P\left(\hat{f}_{V(i-1)}(\theta_m) - f_{H(i)}(\theta_i) > \alpha, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, G_{i-1}(\theta_m,\eta_{i-1}) = 0\right) \\
&= P\left(\hat{f}_{V(i-1)}(\theta_m) - f_{H(i)}(\theta_i) > \alpha, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, \theta_m \in \Theta_{\mathcal{F}}, G_{i-1}(\theta_m,\eta_{i-1}) = 0\right) \\
&\quad + P\left(\hat{f}_{V(i-1)}(\theta_m) - f_{H(i)}(\theta_i) > \alpha, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, \theta_m \in \bar{\Theta}_{\mathcal{F}}, G_{i-1}(\theta_m,\eta_{i-1}) = 0\right) \\
&\leq P\left(\hat{f}_{V(i-1)}(\theta_m) - f(\theta_m) + f(\theta_i) - f_{H(i)}(\theta_i) > \alpha - \epsilon\right) \\
&\quad + P\left(\theta_m \in \bar{\Theta}_{\mathcal{F}}, G_{i-1}(\theta_m,\eta_{i-1}) = 0\right) \\
&\leq P\left(|\hat{f}_{V(i-1)}(\theta_m) - f(\theta_m)| > \frac{\alpha-\epsilon}{2}\right) + P\left(|f(\theta_i) - f_{H(i)}(\theta_i)| > \frac{\alpha-\epsilon}{2}\right) \\
&\quad + P\left(\theta_m \in \bar{\Theta}_{\mathcal{F}}, G_{i-1}(\theta_m,\eta_{i-1}) = 0\right) \\
&\leq \frac{Const}{(i-1)^{c(l-1)}} + \frac{Const}{i^{ql}} + \frac{Const}{\eta_{i-1}^{2w}(i-1)^{c(w-1)}}, \qquad (61)
\end{aligned}
$$

where the first part of the first inequality is due to the fact that $-f(\theta_m) + f(\theta_i) \geq -f^* + f^* - \epsilon = -\epsilon$ when $\theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}$ and $\theta_m \in \Theta_{\mathcal{F}}$. The first term of (61) using Assumption 3.2.1 and the same methodology as the derivation of (9) and (10) of Andradottir and Prudius [13], the second term of (61) is due to Lemma 1 of Andradottir and Prudius [13], and the third term of (61) is due to (32).

We also have

$$(60) \leq \sum_{j \in \mathcal{C}} P\left(|g_{j,H(i)}(\theta_i) - g(\theta_i)| > \beta\right) \leq \frac{Const}{i^{qw}}, \tag{62}$$

due to Lemma 1 of Andradóttir and Prudius [13].

We can see that (59) − (62) yield that

$$P\left(\theta_i \notin \Theta_i, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}\right) \leq \frac{Const}{(i-1)^{c(l-1)-1}} + \frac{Const}{i^{ql-1}} + \frac{Const}{\eta_{i-1}^{2w}(i-1)^{c(w-1)-1}} + \frac{Const}{i^{qw}}. \tag{63}$$

Since $\eta_i = \Omega(i^{-\gamma_\eta})$, $c(l-1) > 2$, $c(w-1) - 2w\gamma_\eta > 2$, $ql > 2$, and $qw > 1$, we have

$$\sum_{i=1}^{\infty} P\left(\theta_i \notin \Theta_i, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}\right) < \infty.$$

From the first Borel-Cantelli lemma, we now know that $P\left(\theta_i \notin \Theta_i, \theta_i \in \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\right) = 0$. Therefore, (58) yields $P\left(\theta_i \in \Theta_i \cap \Theta_{\epsilon,\Delta(\epsilon)}, i.o.\right) = 1$. This completes the proof. □

4.4 Numerical Analysis

The main contribution of this chapter is using penalization and discarding to design a provably convergent algorithm, ASDP, to solve continuous simulation optimization problems with stochastic constraints. In this section, we conduct numerical analysis aimed at investigating how stochastic constraints affect the performance of ASDP. There are four types of constraints we impose on each test problem. The effect of each type of constraints is given in Table 2 (recall that f^* denotes the global optimal objective value of (2), whereas $\bar{f}$ denotes the global optimal value of (2) without stochastic constraints). Both Type I and Type II constraints do not rule out the optimal solution of the corresponding unconstrained problem. Type I constraints indicate the optimal solution is in the interior of the feasible region, whereas Type II constraints indicate that the optimal solution is on the boundary of the feasible region of the constrained problem. Both Type III and Type IV constraints rule out the optimal objective value of the corresponding problem without stochastic constraints.

Table 2: Effects of constraints

Type	Effect
I	$f^* = \bar{f}$ without binding constraints at f^*
II	$f^* = \bar{f}$ with binding constraints at f^*
III	$f^* < \bar{f}$ without binding constraints at f^*
IV	$f^* < \bar{f}$ with binding constraints at f^*

Type III constraints indicate the constrained optimal solution is in the interior of the feasible region, whereas Type IV constraints indicate the constrained optimal solution is on the boundary of the feasible region.

Now we describe our constraints. For s-dimensional problems where $\theta = (x_1, \ldots, x_s)$, we assume the constraints have the following form: for each $j = 1, \ldots, |\mathcal{C}|$,

$$g_j(\theta) = \frac{1}{s} \sum_{i=0}^{s} x_i^j \leq b_j.$$

One main feature of our ASDP algorithm is that the candidate solutions converge to the optimal value from inside the feasible region when $\xi_i > 0$ (rather than convergence from the infeasible region). By contrast, if we set $\xi_i = 0$ for each i in ASDP (call this the ASDP$_0$ algorithm), we actually relax the conditions on penalization, and Theorem 4.3.1 does not guarantee almost sure convergence from inside the feasible region. However, according to Theorem 4.3.2, when $\xi_i = 0$, we can obtain almost sure convergence under Assumption 4.3.4 without guaranteeing convergence from inside the feasible region. To investigate whether restricting the solutions to converge from inside the feasible region will cost algorithm performance or not, we compare the ASDP algorithm with the ASDP$_0$ algorithm under Type II and IV constraints.

The outline of this section is as follows: In Section 4.4.1, we describe our test problems, in Section 4.4.2, we provide implementation details of the tested algorithms, and in Section 4.4.3, we provide, compare, and analyze our numerical results.

4.4.1 Test Problems

This section describes our test problems, namely the Quadratic problem, Two Hills problem, Combined Pinter and Rosenbrock problem, and Combined Griewank and Trigonometric problem. The Quadratic, Two Hills, Pinter, Rosenbrock, Griewank, and Trigonometric problems without stochastic constraints have been used before as test problems in the optimization literature, and most of them are considered in Chapter 3. In this chapter, rather than study all the problems separately as before, we first incorporate stochastic constraints into each test problem, then we combine the Pinter and Rosenbrock problems, and the Griewank and Trigonometric problems together, respectively, to create two new problems to study how our algorithm performs under different types of constraints. The reason we combine the Pinter and Rosenbrock, and Griewank and Trigonometric problems, respectively, is because we would like to test the algorithms on difficult problems with known suboptimal solutions (combination can help us attain these two goals). There are four versions of each test problem representing each type of constraints, respectively.

The first two test problems are low dimensional problems with simple structure, the third one is a 10 dimensional, highly multimodal, and badly scaled problem, and the fourth one is a 20 dimensional problem, also highly multimodal. The constraints are designed to satisfy the requirements in Table 2; under Type III and IV constraints, we set the constraints in a way that Assumption 4.3.4 is satisfied.

The Quadratic (Q-1) problem:

$$f(\theta) = -\theta^2 + 100,$$

$\Theta = [-10, 10]$. The global maximum is 100 at $\theta = 0$, the range of $f(\theta)$ on Θ is $[0, 100]$. Let $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ and $u_j(\theta, Y(\omega)) = g_j(\theta) + Y(\omega)$ for all $\theta \in \Theta$ and $j = 1, \ldots, |\mathcal{C}|$, with $X(\omega)$ being $\mathcal{N}(0, 10)$ and $Y(\omega)$ being $\mathcal{N}(0, 1)$. Let $|\mathcal{C}| = 1$; the right-hand side value b_1 is described in Table 3. The reason we do not have Type

Table 3: Constraints for the Quadratic problem

Type	b_1
I	$b_1 = 5$
II	$b_1 = 0$
IV	$b_1 = -5$

Table 4: Constraints for Two Hills problem

Type	$b_j,\ j = 1, 2, 3$
I	$b_1 = 30,\ b_2 = 1250,\ b_3 = 45000$
II	$b_1 = 27.75,\ b_2 = 1002.625,\ b_3 = 45000$
III	$b_1 = 22.5,\ b_2 = 600,\ b_3 = 30000$
IV	$b_1 = 20,\ b_2 = 500,\ b_3 = 30000$

III constraints is because this problem has no local maximum that is not a global maximum.

The Two Hills (TH-2) problem:

$$f(\theta) = \max\{f_1(\theta), f_2(\theta), 0\},$$

where $\theta = (x_1, x_2)$,

$$f_1(\theta) = -(0.4x_1 - 5)^2 - 2(0.4x_2 - 17.2)^2 + 10,$$

$$f_2(\theta) = -(0.4x_1 - 12)^2 - (0.4x_2 - 4)^2 + 5,$$

and $\Theta = \{(x_1, x_2) \in \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 50\}$. We let $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ for all $\theta \in \Theta$, with $X(\omega)$ being $\mathcal{N}(0, 10)$. This objective function has two hills of different heights (5 and 10), located relatively far apart (the hill of height 5 is centered at $(30, 10)$ and the hill of height 10 is centered at $(12.5, 43)$), and separated by a flat valley (of height 0). The optimal value is $f^* = 10$. Let $|\mathcal{C}| = 3$; the right-hand side values b_j $(j = 1, 2, 3)$ are described in Table 4. We let $u_j(\theta, Y_j(\omega)) = g_j(\theta) + Y(\omega)$ for all $\theta \in \Theta$ and $j = 1, 2, 3$, with $Y(\omega)$ being $\mathcal{N}(0, 1)$.

Table 5: Constraints for the Combined Pinter and Rosenbrock 10D problem

Type	Form
I	$b_1 = 1.9$, $b_2 = 1.9^2$, $b_3 = 1.9^3$
II	$b_1 = 1.5$, $b_2 = 1.5^2$, $b_3 = 1.9^3$
III	$b_1 = 1.25$, $b_2 = 1.25^2$, $b_3 = 1.8^3$
IV	$b_1 = 0$, $b_2 = 1.25^2$, $b_3 = 1.8^3$

The Combined Pinter and Rosenbrock 10D (PR-10) problem:

$$f(\theta) = \begin{cases} f_1(\theta) & \text{if } \theta \in \Theta_L, \\ f_2(\theta) & \text{if } \theta \in \Theta \backslash \Theta_L, \end{cases}$$

where $\theta = (x_1, \ldots, x_s)$, the Pinter problem:

$$\begin{aligned} f_1(\theta) = &- \left(\sum_{i=1}^{s} i x_i^2 + \sum_{i=1}^{s} i \sin^2 \left(x_{i-1} \sin x_i - x_i + \sin x_{i+1} \right) \right) \\ &- \left(\sum_{i=1}^{s} i \log_{10} \left[1 + i (x_{i-1}^2 - 2x_i + 3x_{i+1} - \cos x_i + 1)^2 \right] \right) - 20, \end{aligned}$$

where $x_0 = x_s$, $x_{s+1} = x_1$, the Rosenbrock problem:

$$f_2(\theta) = - \left(\sum_{i=1}^{s-1} \left[(1.5 - x_i)^2 + (1.5 x_{i+1} - x_i^2)^2 \right] + 1 \right),$$

$\Theta = \{(x_1, \ldots, x_s) \in \mathbb{R}^s : -2 \leq x_i \leq 2, i = 1, \ldots, s\}$, and $\Theta_L = \{(x_1, \ldots, x_s) \in \mathbb{R}^s : \frac{1}{s} \sum_{i=1}^{s} x_i^2 \leq 1.3^2\}$. We assume $s = 10$. Note that $f(\theta) = f_1(\theta)$ on Θ_L with a local maximum -20 at $(0, \ldots, 0)$, the approximate range of $f_1(\theta)$ on Θ_L is $[-20, -500)$, and $f(\theta) = f_2(\theta)$ on $\Theta \backslash \Theta_L$ with a local maximum -1 at $(1.5, \ldots, 1.5)$, the approximate range of $f_2(\theta)$ on $\Theta \backslash \Theta_L$ is $[-1, -700)$. Let $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ for all $\theta \in \Theta$, with $X(\omega)$ being $\mathcal{N}(0, 100)$. Let $|\mathcal{C}| = 3$; the right-hand side values b_j $(j = 1, 2, 3)$ are described in Table 5. Let $u_j(\theta, Y_j(\omega)) = g_j(\theta) + Y(\omega)$ for all $\theta \in \Theta$ and $j = 1, 2, 3$, with $Y(\omega)$ being $\mathcal{N}(0, 1)$.

The Combined Griewank and Trigonometric 20D (GT-20) problem:

$$f(\theta) = \begin{cases} f_1(\theta) & \text{if } \theta \in \Theta_L, \\ f_2(\theta) & \text{if } \theta \in \Theta \backslash \Theta_L, \end{cases}$$

Table 6: Constraints for the Combined Griewank and Trigonometric 20D problem

Type	Form
I	$b_1 = 4.9$, $b_2 = 4.9^2$, $b_3 = 4.9^3$
II	$b_1 = 1.5$, $b_2 = 4.9^2$, $b_3 = 4.9^3$
III	$b_1 = 0.75$, $b_2 = 4.8^2$, $b_3 = 4.8^3$
IV	$b_1 = -1$, $b_2 = 4.8^2$, $b_3 = 4.8^3$

where $\theta = (x_1, \ldots, x_s)$, the Griewank problem:

$$f_1(\theta) = -\left(\sum_{i=1}^{s}(x_i+1)^2 - \prod_{i=1}^{s}\cos(\frac{x_i+1}{\sqrt{i}})\right) - 21,$$

and the Trigonometric problem:

$$f_2(\theta) = -\left(\frac{1}{20}\sum_{i=1}^{s}(|x_i - 1.5|)^3 + \sum_{i=1}^{s}\sin^2(7(x_i-1.5)^2)\right) - 1,$$

$\Theta = \{(x_1, \ldots, x_s) \in \mathbb{R}^s : -5 \leq x_i \leq 5, i = 1, \ldots, s\}$, and $\Theta_L = \{(x_1, \ldots, x_s) \in \mathbb{R}^s : \frac{1}{s}\sum_{i=0}^{s} x_i \leq 1\}$. We assume $s = 20$. Note that $f(\theta) = f_1(\theta)$ on Θ_L with a local maximum -20 at $(-1, \ldots, -1)$, the approximate range of $f_1(\theta)$ on Θ_L is $[-20, -500)$, and $f(\theta) = f_2(\theta)$ on $\Theta \backslash \Theta_L$ with a local maximum -1 at $(1.5, \ldots, 1.5)$, the approximate range of $f_2(\theta)$ on $\Theta \backslash \Theta_L$ is $[-1, -300)$. Let $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ for all $\theta \in \Theta$, with $X(\omega)$ being $\mathcal{N}(0, 100)$. Let $|\mathcal{C}| = 3$; the right-hand side values b_j $(j = 1, 2, 3)$ are described in Table 6. Let $u_j(\theta, Y_j(\omega)) = g_j(\theta) + Y(\omega)$ for all $\theta \in \Theta$ and $j = 1, \ldots, s$, with $Y(\omega)$ being $\mathcal{N}(0, 1)$.

Now we discuss how we choose the noise for each problem. We make the noise reasonably large for both the objective function and constraints. Here "reasonable" means the objective function noise should not be too large compared to the range of objective function values. Moreover, if we define the diameter of some closed set S as the maximum infinity norm of two points in S, the constraints noise should not be too large compare to the diameter of the sampling region Θ. Therefore, we choose the variance of the objective function noise to be 10 for the Q-1 and TH-2 problems and 100 for the PR-10 and GT-20 problems. For constraints noise, we set the variance to be 1 for all test problems.

4.4.2 Algorithm Implementation

In this section, we provide implementation details for the ASDP algorithm. For the sampling strategy, we use the balanced exploration and exploitation approach to sample new points; the original idea can be found in Andradóttir and Prudius [12]. Explicitly, in iteration $k = V(i)$, with probability $p > 0$, a new solution is sampled uniformly from the whole feasible set Θ, and with probability $1 - p$, if $\hat{g}_{j,V(i-1)}(\theta^*_{i-1}) \leq b_j - \eta_i/s$ for any $j \in \mathcal{C}$, then, a new solution is sampled uniformly from $N(\theta^*_{i-1})$, where

$$N(\theta) = N((x_1, \ldots, x_s)) = \{(x'_1, \ldots, x'_s) \in \Theta : |x_i - x'_i| \leq r, i = 1, \ldots, s\}$$

for all $\theta \in \Theta$, otherwise, a new solution is still sampled uniformly from the whole region (the first point is sampled uniformly from Θ). Here we use r to denote the radius of the "local" neighborhood. The sampling approach is similar to that used for implementing ASR of Andradóttir and Prudius [13] and ASRD in Chapter 3, except here we conduct feasibility test before doing local search. The main reason is to avoid performing local search around the infeasible region. (Notice that ASR and ASRD are applied to solve simulation optimization problems with "simple" (for example, simplex, integer lattice, etc.) feasible regions, whereas ASDP is aimed at solving simulation optimization problem with stochastic constraints.) After a new point is sampled, we use the acceptance criterion described in Section 4.3.5 to decide whether to keep or abandon the newly sampled point. We choose $p = 0.5$ for all test problems.

Next, we describe our resampling strategy and the acceptance criterion. The purpose of the resampling procedure is to reduce the noise to get more accurate estimates of the objective (and constraint) function values. However, whether it is desirable to conduct resampling or not depends on the properties of the underlying objective function, as well as the magnitude of the noise in the objective function

values. A brief discussion of how to determine whether resampling is beneficial or not can be found in Section 3.3.4. Here, we incorporate resampling in our ASDP implementation since we set the noise big enough for both the objective functions and constraints so that resampling is desirable. We use the same resampling strategy as in Section 3.3.2. The implementation details are as follows. Let $V(i) = \lfloor i^v \rfloor$, where $v \geq 1$, and note that $m_k = \lfloor k^{1/v} \rfloor$ is the number of points sampled by the end of iteration k. Then, a point $\theta \in \Theta^*_{m_k}$ is resampled in iteration k with probability

$$p_k(\theta) = \frac{\exp\left(\hat{F}_{m_k}(\theta)\right)}{\sum_{\theta' \in \Theta^*_{m_k}} \exp\left(\hat{F}_{m_k}(\theta')\right)},$$

where $\hat{F}_{m_k}(\theta) = \min\{\max\{\underline{U}, F_{m_k}(\theta)/T\}, \overline{U}\}$, with $\overline{U} > \underline{U} > 0$ and $T > 0$. Here we choose $\overline{U} = 400$ and $\underline{U} = -400$.

Finally, we discuss how to choose parameter values. For our ASDP algorithm, choose $\delta_i = D_\delta / i^{\gamma_\delta}$, $\eta_i = 1.01\xi_i$, and $\xi_i = D/i^\gamma$, where $\gamma_\delta > 0$ and $\gamma > 0$. For the ASDP$_0$ algorithm, choose $\delta_i = D_\delta / i^{\gamma_\delta}$, $\eta_i = 1.01 D/i^\gamma$, and $\xi_i = 0$. Let $\gamma = 0.24$, and $D = \beta = 1$ for all test problems. Let $D_\delta = \sqrt{10}$, and $\gamma_\delta = 0$ for Q-1 problem. Let $D_\delta = \sqrt{10}$ and $\gamma_\delta = 0.2$ for TH-2 problem. Let $D_\delta = 10$, and $\gamma_\delta = 0.2$ for PR-10, and GT-20 problems. Here we choose D_δ be the standard deviation of the noise in objective function, and D_δ, D, and β to be the standard deviation of the noise in the stochastic constraints (in practice, these values would need to be estimated). Let $\lambda_i = i^\rho$, $K(i) = \lceil S i^c \rceil$, and $H(i) = \lceil Q i^q \rceil$, where $\rho, S, c, Q, q > 0$, and choose $\rho = 0.5$, $c = 0.5$, $Q = 1$, $q = 0.05$, and $v = 1.1$. Choose $S = 5$ for Q-1 and TH-2 problems, $S = 1$ for PR-10 and GT-20 problems. Notice here we choose $\lambda_i = i^\rho$ for $\rho > 0$ as $\lim_{i\to\infty} \lambda_i = \infty$ to satisfy the condition: $\liminf_{i\to\infty} \lambda_i > \bar{f} - f^*$. Since the noise follows normal distributions, which corresponds to $l, w = \infty$; for all of the test problems, we only need $c > 2\gamma_\delta$, $c > 2\gamma_\eta$, $c > 2\gamma_\xi$ and $q > 0$. Moreover, we choose $r = 0.2$ for Q-1 problem, $r = 0.5$ for TH-2 problem, $r = 0.04$ for PR-10 problem, and $r = 0.1$ GT-20 problem, here r is chosen to be $\frac{1}{100}$ of the diameter of the feasible

region for each test problem. The additional parameters for acceptance criterion and resampling are $\alpha = 0.1$ for all test problems, $T = 0.1$ for Q-1 and TH-2 problems, and $T = 100$ for PR-10 and GT-20 problems. We take five objective function observation on each resampling step. Notice here we choose η_i to be the same in both ASDP and ASDP$_0$ algorithms to compare how feasibility guarantee affects the performance. Also, most of the parameter values are as in Andradóttir and Prudius [13] and Chapter 3, since some steps of ASDP are similar to ASRD. How to determine a priori the most appropriate values of all the parameters is outside of the scope of this thesis.

Let $N_k = \sum_{\theta \in \tilde{\Theta}_{m_k}} N_k(\theta)$ be the total number of objective function evaluations by the end of iteration k. Let N be the simulation budget. The performance of the algorithms is averaged over $W = 100$ independent replications for all test problems. Their performance is documented by plotting 100 pairs (x, y), where $x \in \{0.01N, 0.02N, \ldots, N\}$, and y is the average objective function value at the estimated optimal solution after x objective function observations have been collected. As the estimate of the optimal solution is only updated in iterations $V(1)$, $V(2)$,..., the value of y is the same for all corresponding $x \in [N_{V(i)}, N_{V(i+1)})$. Moreover, for each k, define $R^*_{m_k} = \sum_{i=1}^{W} \mathbb{1}_{\{\sum_{j \in \mathcal{C}} \mathbb{1}_{\{g_j(\theta^*_{m_k}) > b_j\}} = 0\}} / W$, which is the proportion over W independent replications of the estimated optimal solutions after k objective function observations have been collected that are feasible.

4.4.3 Performance Comparison

Figures 11, 14, 17, and 20 show the empirical performance of the ASDP method under four different types of constraints on the Quadratic (Q-1), Two Hills (TH-2), combined Pinter and Rosenbrock 10D (PR-10), and combined Griewank and Trigonometric 20D (GT-20) problems, respectively. Figures 12, 15, 18, and 21 show the empirical performance of the ASDP and ASDP$_0$ methods under Type II and IV constraints. Figures 13, 16, 19, and 22 show the empirical feasibility performance of the ASDP

and ASDP_0 methods under Type II and IV constraints. Moreover, the horizontal line labeled "$\bar{f}$" denotes the optimal value of the objective function under Type I and II constraints, whereas the other horizontal line labeled "f^*" denotes the optimal value of the objective function under Type III and IV constraints. We plot $-f(\theta^*_{m_k})$ for the PR-10 and GT-20 problems (as both functions have negative objective function values); therefore smaller values are better for Figures 17, 18, 20, and 21 (larger values are better for Figures 11, 12, 14, and 15). For Figures 13, 16, 19, and 22, higher values indicates that the estimate of the optimal solution is more likely to be feasible. Finally, the sequence in which the numerical results are presented moves from lower dimensional, smoother problems to higher dimensional problems with greater curvature. In the following, we will analyze these four problems in detail.

For the Q-1 problem, from Figure 11 it is clear that the ASDP algorithm converges under Type I and Type II constraints. Although the optimal solution is the same for Types I and II, the convergence rate is slower for Type II due to the difficulty of ensuring convergence from inside the feasible region when the optimal solution is on the boundary of the feasible region. Moreover, under Type IV constraints (optimal solution is on the boundary), although the objective function values of infeasible points that are near the boundary are larger than the optimal value, we can see from Figure 11 that ASDP converges from inside the feasible region. From Figure 12, we can see that under Type II constraints, ASDP_0 has slightly better performance over ASDP. Under Type IV constraints, ASDP_0 performs much better than ASDP. The main reason is, since ASDP_0 does not guarantee convergence from inside the feasible region, and the objective function values of infeasible points that are near the boundary are larger than the optimal value, ASDP_0 chooses both feasible and infeasible points around the boundary (as demonstrated in Figure 13). From Figure 13, we can see that ASDP selects much more feasible solutions as the estimate of the optimal solution, and from the numerical results in Figure 12, we know that it costs

performance to guarantee convergence from inside the feasible region.

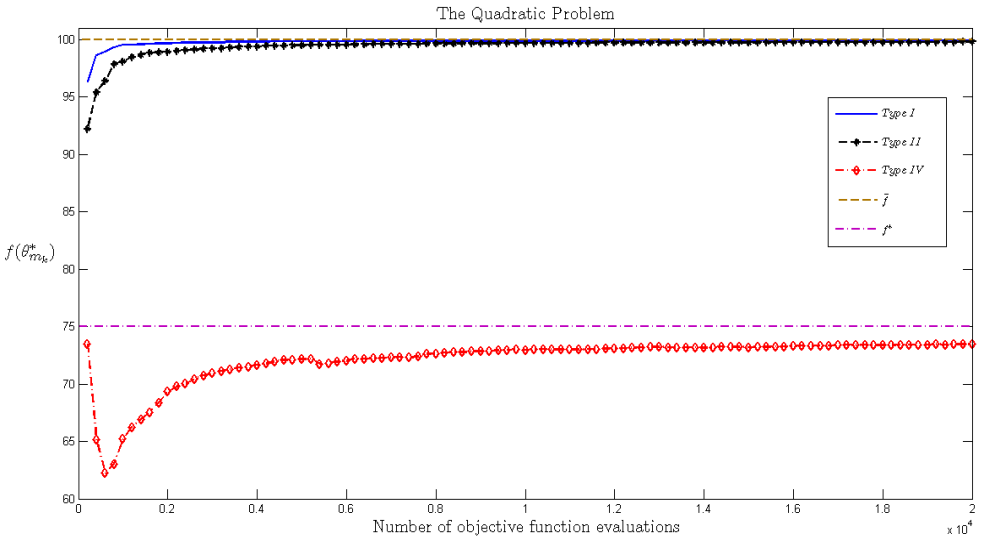

Figure 11: Performance of the ASDP method on the Quadratic problem under different types of constraints

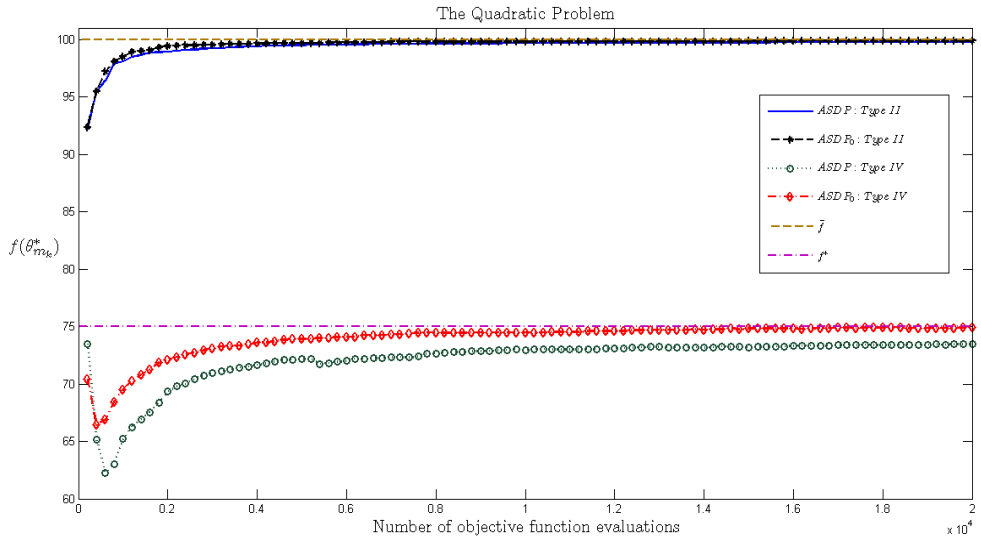

Figure 12: Performance of the ASDP and ASDP$_0$ methods on the Quadratic problem under Type II and Type IV constraints

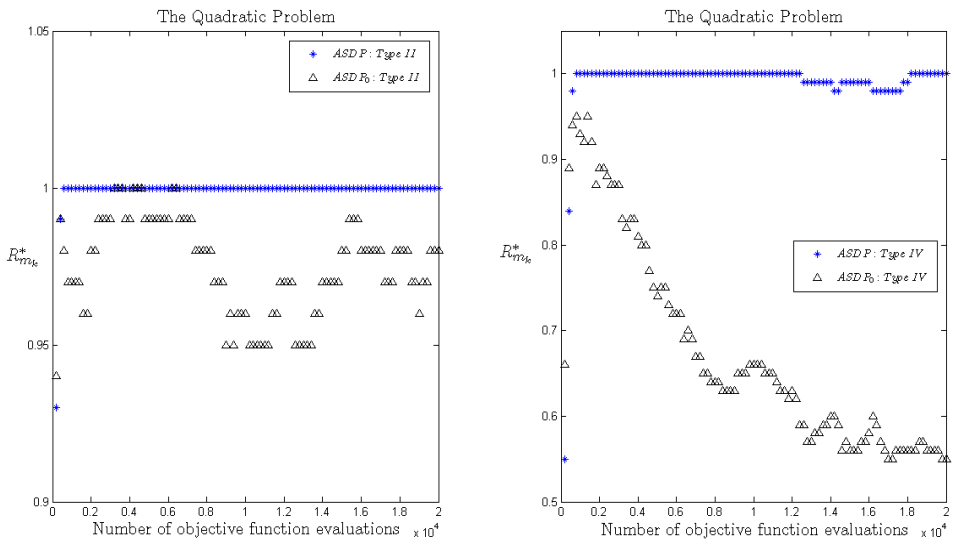

Figure 13: Feasibility performance of the ASDP and ASDP_0 methods on the Quadratic problem under Type II and Type IV constraints

For the TH-2 problem, from Figure 14 it is evident that the ASDP algorithm convergences under all types of constraints. The convergence rate is slower for Types II and IV compared to Types I and III, respectively, due to ensuring convergence from inside the feasible region. From Figure 15, we notice that ASDP_0 and ASDP have almost the same performance under both Type II and IV constraints. Moreover, Figure 16 demonstrates that ASDP guarantees convergence from inside the feasible region whereas ASDP_0 fails to do. But we also notice that at least 95% of the estimates of the optimal solution are feasible under ASDP_0, which is promising.

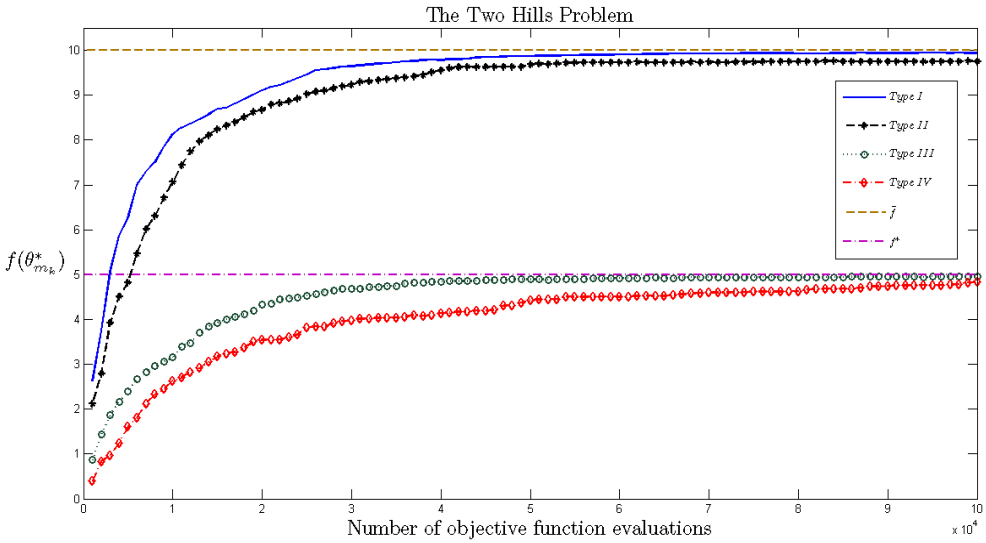

Figure 14: Performance of the ASDP method on the Two Hills problem under different types of constraints

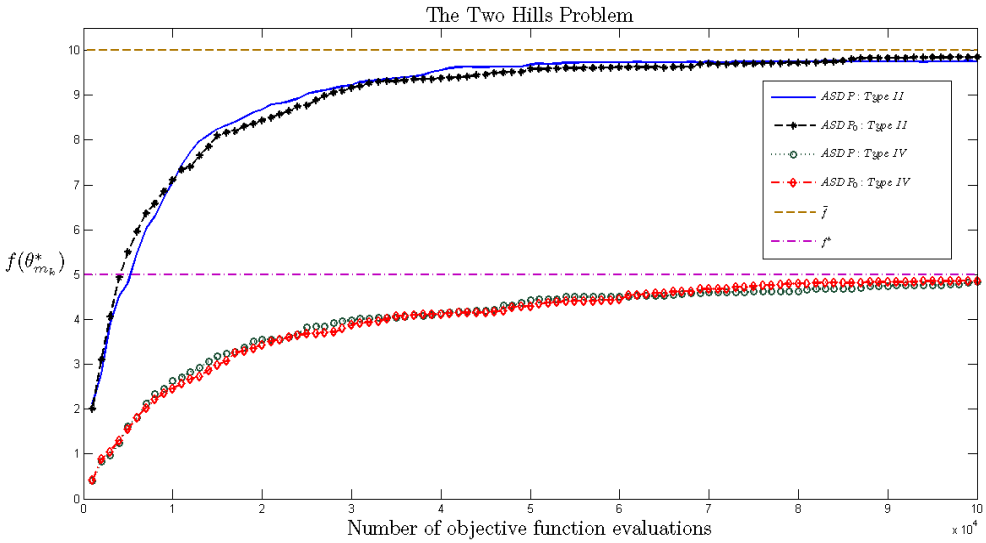

Figure 15: Performance of the ASDP and ASDP$_0$ methods on the Two Hills problem under Type II and Type IV constraints

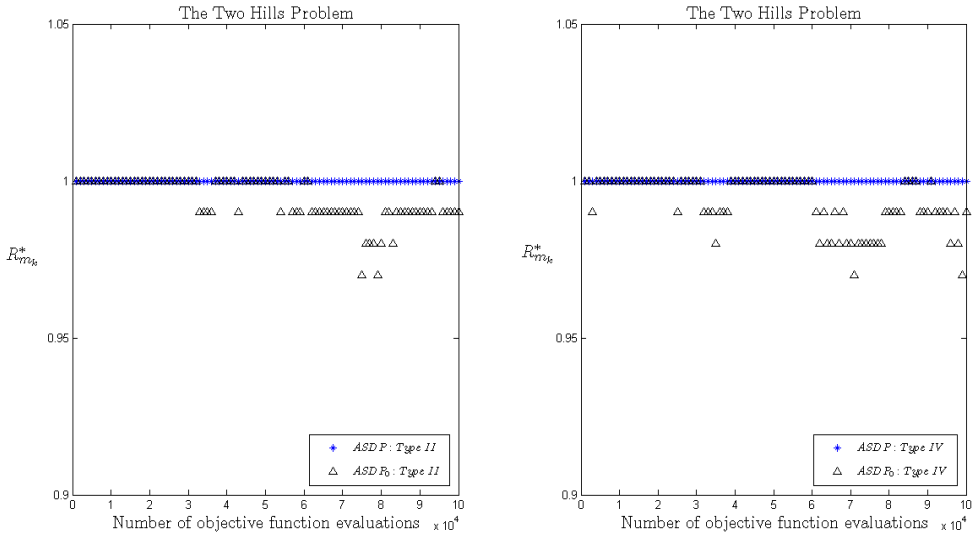

Figure 16: Feasibility performance of the ASDP and ASDP$_0$ methods on the Two Hills problem under Type II and Type IV constraints

Next, we analyze the figures for the high dimensional problems. For the PR-10 problem (Figure 17), as demonstrated in the lower dimensional problems, ASDP is able to identify feasible and promising solutions under different types of constraints. Especially, under Type III and IV constraints where the constrained optimal value is less than the unconstrained optimal value, the algorithm finds the right path to converge instead of seeking solutions with higher objective values but infeasible. As before, the convergence rate is slower under Type II and IV constraints than Types I and III, due to the difficulty of ensuring convergence from inside the feasible region when the optimal solution is on the boundary of the feasible region. Also, we can see that ASDP performs much better under Type III constraints than Type IV constraints. The main reason is that since the dimension is high, the volume of the feasible region compared to the volume of the whole sampling space under Type IV constraints is much smaller than for Type III constraints, and it is more difficult to sample a point from the feasible region using adaptive search under Type IV constraints than Type

III constraints. From Figure 18, it is evident that it costs performance to converge from inside the feasible region; hence ASDP_0 performs better than ASDP under both Type II and IV constraints. Finally, the results in Figure 19 support our theory that ASDP guarantees convergence from inside the feasible region. Although ASDP_0 has noticeably better performance than ASDP under Type IV constraints, the estimate of the optimal solution has about 15% chance to be infeasible.

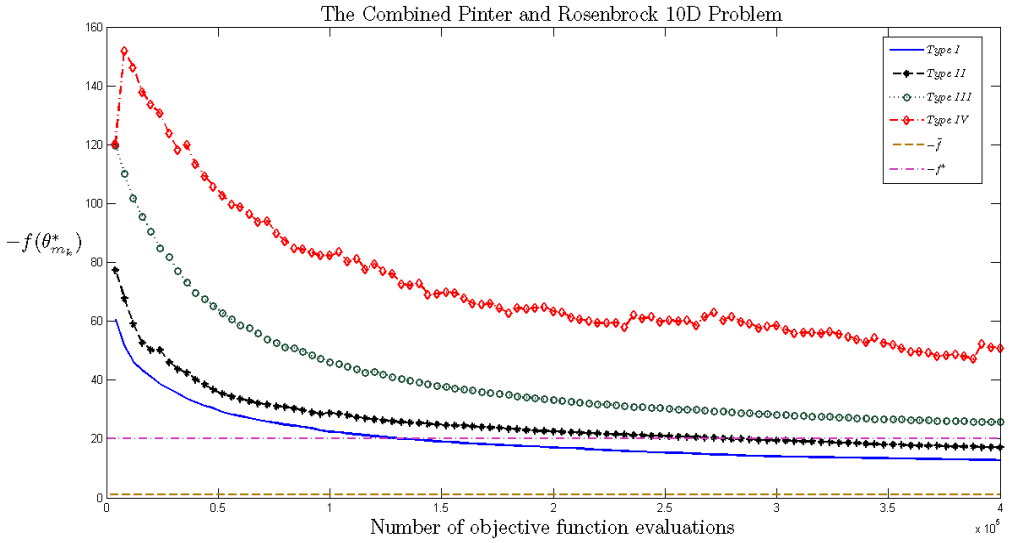

Figure 17: Performance of the ASDP method on the Combined Pinter and Rosenbrock 10D problem under different types of constraints

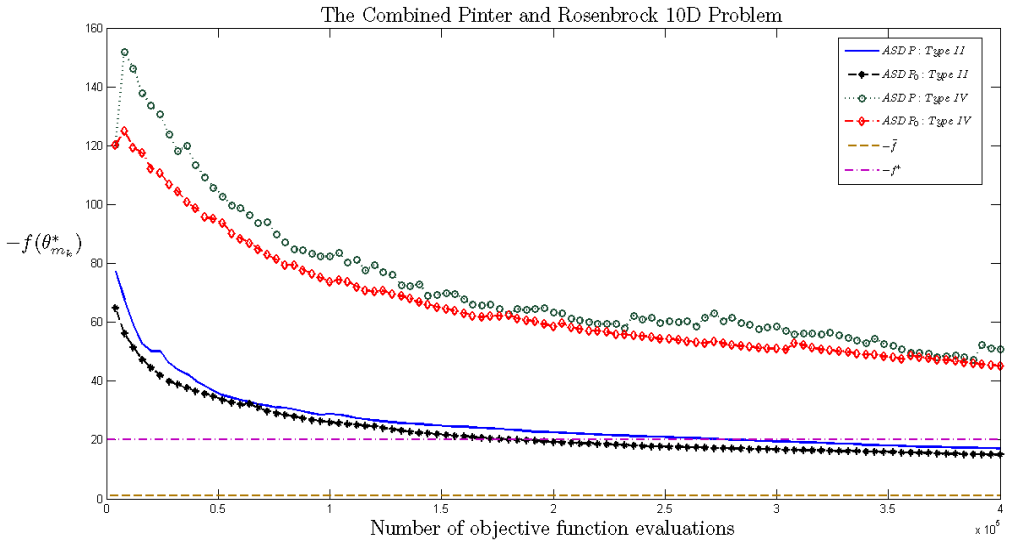

Figure 18: Performance of the ASDP and ASDP$_0$ methods on the Combined Pinter and Rosenbrock 10D problem under Type II and Type IV constraints

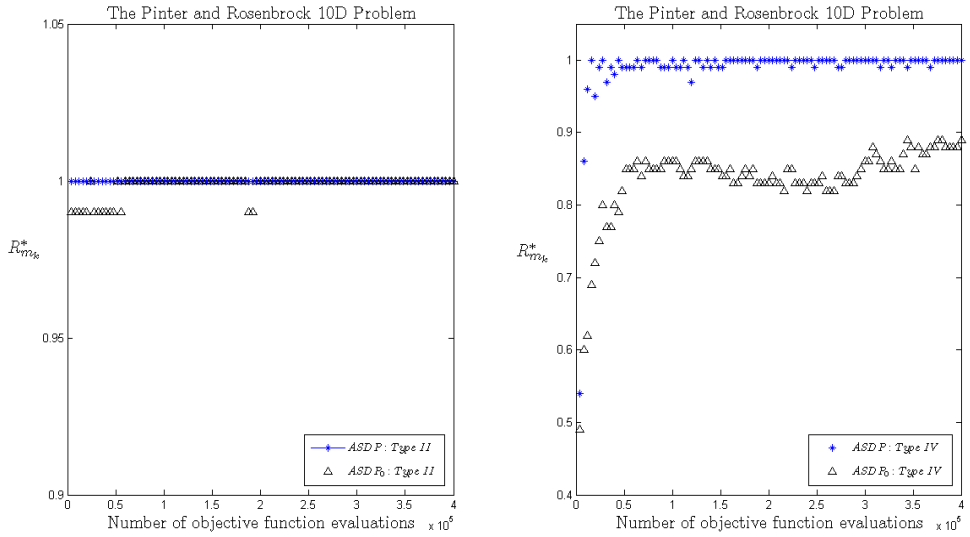

Figure 19: Feasibility performance of the ASDP and ASDP$_0$ methods on the Combined Pinter and Rosenbrock 10D problem under Type II and Type IV constraints

For the GT-20 problem, from Figure 20 the algorithm performs similarly as for the

combined PR-10 problem (Figure 17), except that here the performance difference of ASDP under Type III and IV constraints is even bigger. One reason is that the optimal solution lies on the boundary and it is even more difficult to guarantee convergence from inside the feasible region without hurting performance than for the PR-10 problem, since the dimension here is twice as high as for the PR-10 problem. From Figure 21, we know that ASDP_0 has similar performance as ASDP under Type II constraints. However, ASDP_0 has much better performance over ASDP under Type IV constraints; the reason is similar as for the PR-10 problem. However, from Figure 22, we can see clearly that although ASDP and ASDP_0 have similar performance under Type II constraints, ASDP does guarantee the convergence from inside the feasible region, whereas ASDP_0 does not have this feature.

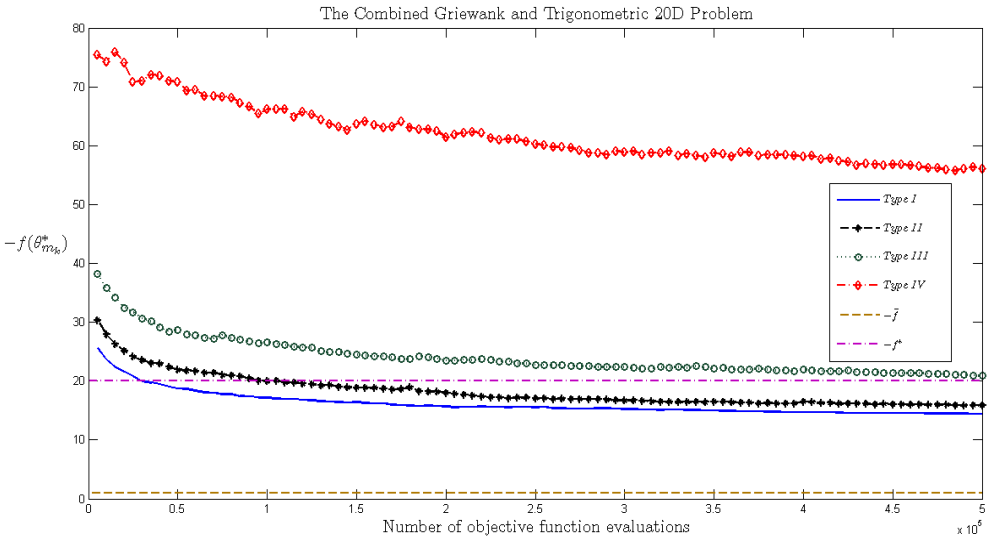

Figure 20: Performance of the ASDP method on the Griewank and Trigonometric 20D problem under different types of constraints

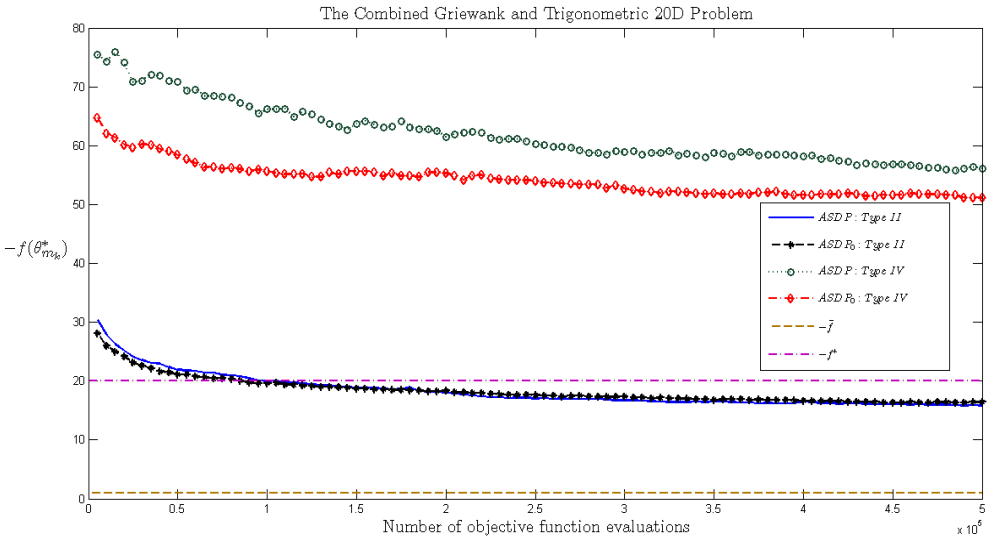

Figure 21: Performance of the ASDP and ASDP_0 methods on the Griewank and Trigonometric 20D problem under Type II and Type IV constraints

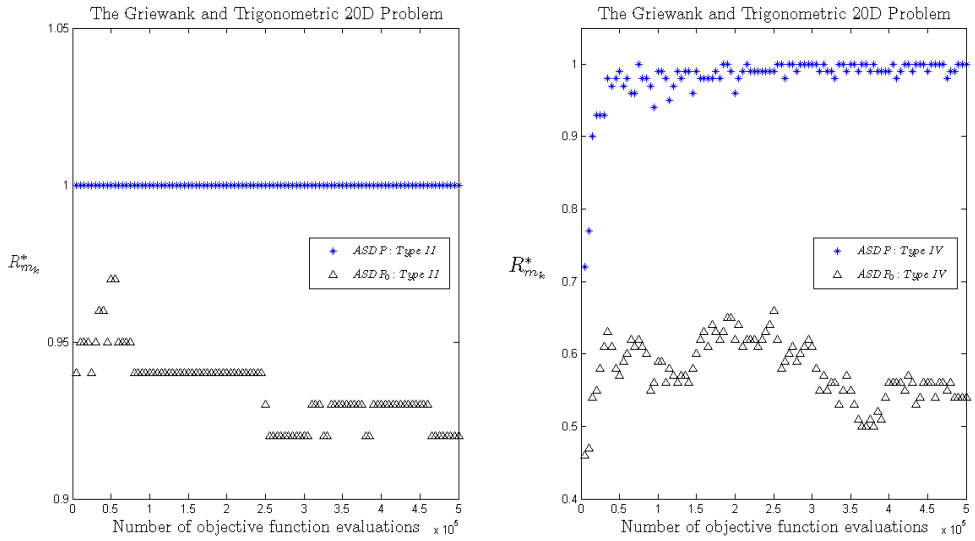

Figure 22: Feasibility performance of the ASDP and ASDP_0 methods on the Griewank and Trigonometric 20D problem under Type II and Type IV constraints

4.5 Conclusion

In this chapter, we designed and analyzed a stochastic search algorithm, named Adaptive Search with Discarding and Penalization (ASDP), to solve continuous simulation optimization problems with stochastic constraints. The method is shown to converge almost surely from inside the feasible region under mild conditions on algorithm parameters and the underlying problem. We also provide conditions under which the algorithm converges to the optimal solution almost surely, but without necessarily converging from inside the feasible region. We provide numerical results showing that ASDP does indeed guarantee convergence from inside the feasible region and that sometimes it costs performance to guarantee convergence from inside the feasible region, compared to the version without feasibility guarantee.

CHAPTER V

GAUSSIAN SEARCH WITH RESAMPLING AND DISCARDING FOR CONTINUOUS SIMULATION OPTIMIZATION

5.1 Introduction

In this chapter, we design a sampling distribution based on a Gaussian process and combine it with the ASRD framework of Chapter 3 to develop an algorithm for solving optimization problems involving continuous decision variables and uncertainties. Our sampling distribution is motivated by the work of Sun, Hong, and Hu [65] who considered discrete simulation optimization.

In each sampling iteration of a random search algorithm, a sampling strategy is needed, and new solutions are selected based on the sampling strategy. While implementing ASRD in Chapter 3, only the current estimate of the optimal solution was used to guide the local exploitation. Therefore this implementation of ASRD does not utilize the entire sampled population to conduct the sampling. In this chapter, rather than solely relying on the current estimate of the optimal solution, we utilize the sampled population (no matter whether the points are promising or not) to construct an adaptive sampling distribution. The convergence results for the ASRD framework are not affected by the specific sampling strategy and acceptance criterion, as long as Assumption 3.2.3 is satisfied. This means that if the adaptive sampling distribution is carefully constructed, almost sure convergence still holds.

This chapter is organized as follows: In Section 5.2, we review the fast construction of a Gaussian process by Sun, Hong, and Hu [65], and construct a new Gaussian process that can be combined with the ASRD framework in continuous space. In

Section 5.3, we provide a detailed description of our GSRD framework. In Section 5.4, we discuss the needed assumptions, and prove the almost sure convergence of GSRD. In Section 5.5, we provide a numerical study aimed at comparing the performance of GSRD (which has a model-based sampling strategy) and ASRD with a point-based sampling strategy. Finally, in Section 5.6, we summarize the main contributions of the chapter.

5.2 Gaussian Process-Based Sampling

The objective of this section is to develop a sampling strategy for continuous simulation optimization problems that involves approximating the objective function via a Gaussian process. The outline of this section is as follows. In Section 5.2.1, we review the Gaussian process constructed by Sun, Hong, and Hu [65] for discrete optimization. In Section 5.2.2, we extend their model into continuous space, and construct a new Gaussian process.

5.2.1 Fast Construction of a Gaussian Process by Sun, Hong, and Hu [65]

Sun, Hong, and Hu [65] proposed a Gaussian Process-based random Search (GPS) method to solve discrete simulation optimization problems. Their approach derives a sampling distribution from a Gaussian process based on previously evaluated solutions. Their novel sampling distribution has the property that it can automatically balance the tradeoff between exploitation and exploration. The construction of a sampling distribution is based on the prior belief that the solutions around good solutions tend to be good, whereas the solutions around bad solutions tend to be bad. It has the following desired properties:

- Allocating higher probabilities around better solutions than inferior solutions;
- Allocating higher probabilities to less explored regions than fully explored regions;

- Allocating higher probabilities to less explored regions than around inferior solutions.

Sun, Hong, and Hu [65] assume that Θ is a finite set, and, moreover, $\Theta = \Omega \cap \mathcal{Z}^d$, where $\Omega \subset \mathbb{R}^d$ is a convex, compact set and $\mathcal{Z}^d$ is the set of d-dimensional integer vectors (here we use different notation for the objective function, feasible solutions, and certain other quantities).

For all $\theta \in \Theta$ and $n \in \mathbb{N}^+$, let $f_n(\theta)$ denote the sample mean calculated from n observations of $f(\theta)$. Suppose that, through the current iteration, a random search algorithm has visited m points, denoted as $\theta_1, \ldots, \theta_m$, and has taken n_i simulation replications for θ_i, $i = 1, \ldots, m$. Suppose that $M(\theta)$ is a stationary Gaussian process with mean 0 and covariance function $\sigma^2\gamma(\cdot, \cdot)$, where σ is a positive number. For any $\theta, \theta' \in \Theta$, let $||\cdot||$ denote the Euclidean distance. Suppose that $\gamma(\theta, \theta') = Corr(M(\theta), M(\theta'))$ is a function of $||\theta - \theta'||$, denoted by $h(||\theta - \theta'||)$, where the correlation function $h(\cdot)$ satisfies the following conditions: $0 \leq h(t) \leq 1$ is a decreasing function of t when $t \geq 0$ and, for any $\theta_0, \theta_1, \theta_2 \in \Theta$, $h(||\theta_1 - \theta_2||) \geq h(||\theta_0 - \theta_1||) \times h(||\theta_0 - \theta_2||)$. For instance, $h(t) = e^{-at^2}$ for some $a > 0$ satisfies the condition. Moreover, let $\epsilon(\theta)$ be a $\mathcal{N}(0, \sigma^2(\theta))$ random variable, where $\mathcal{N}(\mu, \sigma^2)$ denotes a normal distribution with mean μ and variance σ^2. We assume $Cov(\epsilon(\theta), \epsilon(\theta')) = 0$ for any $\theta \neq \theta'$.

Based on the sampled points $\theta_1, \ldots, \theta_m$ and the objective function estimations at these points, Sun, Hong, and Hu [65] model $f(\theta)$ via $Y(\theta)$ defined as follows:

$$Y(\theta) = M(\theta) + \lambda(\theta)^T(\bar{\mathbb{F}} - \mathbb{M}) + \lambda(\theta)^T\mathcal{E}, \tag{64}$$

where $\bar{\mathbb{F}} = (\max\{f_{n_1}(\theta_1), \underline{M}\}, \ldots, \max\{f_{n_m}(\theta_m), \underline{M}\})^T$, $\underline{M}$ is a very small negative number (usually -10^{10}) (the parameter $\underline{M}$ prevents $F(\cdot)$ from going to negative infinity, which makes it hard to utilize $Y(\theta)$ to construct sampling distribution), $\lambda(\theta) = (\lambda_1(\theta), \ldots, \lambda_m(\theta))^T$ is a vector of weight functions, $\mathbb{M} = (M(\theta_1), \ldots, M(\theta_m))^T$

is a vector of $M(\theta)$ evaluated at $\theta_1, \ldots, \theta_m$, and $\mathcal{E} = (\epsilon(\theta_1), \ldots, \epsilon(\theta_m))^T$ is an m-dimensional random vector following a multivariate normal distribution with mean 0 and covariance matrix:

$$\Sigma_{\mathcal{E}} = \mathbf{diag}\left\{\frac{\sigma(\theta_1)^2}{n_1}, \ldots, \frac{\sigma(\theta_m)^2}{n_m}\right\}.$$

Furthermore, $\bar{\mathbb{F}}$, $\mathbb{M}$, and $\mathcal{E}$ are mutually independent of each other.

The main idea of equation (64) (refer to Sun, Hong, and Hu [65]) is to use three parts to model the different aspects of $f(\theta)$ as follows:

(i) The model uses a stationary Gaussian process $M(\theta)$ to capture the continuity and the uncertainty of $f(\theta)$ when there is no additional information;

(ii) It incorporates the information from past observations at $\theta_1, \ldots, \theta_m$ in the term $\lambda(\theta)^T(\bar{\mathbb{F}} - \mathbb{M})$;

(iii) It captures the randomness in $f_{n_i}(\theta_i)$, $i = 1, \ldots, m$, by the term $\lambda(\theta)^T\mathcal{E}$ (thus common random numbers are not allowed here due to zero correlation between different points). In fact, Sun, Hong, and Hu [65] estimate $\sigma^2(\theta_i)$ via $\max\{\hat{\sigma}^2(\theta_i), \sigma_0^2\}$, where $\hat{\sigma}^2(\theta_i)$ is the sample variance of n_i observations of $f(\theta_i)$, and σ_0^2 is a predetermined small positive constant.

Let $\mathcal{G}_n$ denote the σ-algebra generated by the sampled points and objective function observations associated with them by the end of iteration n for $n \in \mathbb{N}^+$. In the following, without special notice, $E^*(\cdot)$, $Var^*(\cdot)$, and $P^*(\cdot)$ denote expectation, variance, and probability conditioned on $\mathcal{G}_n$, where n is the most recently completed iteration. Sun, Hong, and Hu [65] propose a sampling distribution as follows: Suppose that the best estimated objective function value found so far is $\hat{f}^*$. Then define the sampling distribution as:

$$g(\theta) = \frac{P^*(Y(\theta) > \hat{f}^*)}{\sum_{\theta' \in \Theta} P^*(Y(\theta') > \hat{f}^*)}, \tag{65}$$

where, for any $\theta \in \Theta$, $P^*(Y(\theta) > \hat{f}^*)$ denotes the conditional probability that the objective function value at θ is better than $\hat{f}^*$. Thus the sampling distribution $g(\cdot)$ displays the relative importance of θ among all solutions in Θ in terms of the conditional probability of being a better solution than the current best.

To ensure the Gaussian model (64) produces a sampling distribution with the desired properties listed at the beginning of this section, Sun, Hong, and Hu [65] require the following conditions on the weight function $\lambda(\theta)$: for any $\theta \in \Theta$, $\lambda(\theta)$ is continuous in θ and satisfies:

(i) $\lambda_i(\theta) \geq 0$ for any $i = 1, \ldots, m$;

(ii) $\sum_{i=1}^{m} \lambda_i(\theta) = 1$; and

(iii) $\lambda_i(\theta_j) = \mathbb{1}_{\{\theta_i = \theta_j\}}$, for all $i, j = 1, \ldots, m$, where $\mathbb{1}_A$ is 1 if event A is true, 0 otherwise.

For instance, one definition Sun, Hong, and Hu [65] suggested is:

$$\lambda_i(\theta) = \begin{cases} \frac{||\theta - \theta_i||^{-b}}{\sum_{j=1}^{m} ||\theta - \theta_j||^{-b}} & \text{if } \theta \neq \theta_i, \\ 1 & \text{if } \theta = \theta_i, \end{cases}$$

for some $b > 0$.

Sun, Hong, and Hu [65] incorporated the Gaussian distribution derived from (64) based on previously evaluated solutions into a random search algorithm (called Gaussian Process-based Search, GPS), and showed the GPS algorithm converges almost surely as the number of iterations goes to infinity. However, GPS only applies to simulation optimization problems with finite decision spaces.

5.2.2 Gaussian Sampling for Continuous Space

The objective in this section is to construct a sampling distribution that balances exploitation and exploration for optimization problems with continuous decision spaces. We will extend the Gaussian process of Sun, Hong, and Hu [65] to construct a new

Gaussian process that can be combined with the ASRD framework to solve problems in continuous decision space.

As in Chapter 3, for all $\theta \in \Theta$ and $k \in \mathbb{N}$, let $N_k(\theta)$ be the number of objective function observations collected at θ by the end of iteration k and let $S_k(\theta)$ be the sum of these $N_k(\theta)$ objective function observations. Also, for all $\theta \in \Theta$ and $k \in \mathbb{N}$, let $\hat{f}_k(\theta) = S_k(\theta)/N_k(\theta)$.

Let $\{V(i)\}$ be a strictly increasing sequence of positive integers with $V(1) = 1$, and let $\{\xi_i\}$ and $\{\eta_i\}$ be two positive real-number sequences. Suppose that, at the end of iteration $V(i)$, we have already sampled a set of points, denoted by $\tilde{\Theta}_i$. Choose $m(i)$ points from $\tilde{\Theta}_{i-1}$, denoted as $\theta_{i,1}, \ldots, \theta_{i,m(i)}$, and note that $m(i)$ may be random. Whereas Sun, Hong, and Hu [65] always use all available points (in which case $m(i) = |\tilde{\Theta}_{i-1}|$), where $|A|$ denotes the cardinality of set A, we choose $m(i) \leq |\tilde{\Theta}_{i-1}|$ points instead of using all the available points. The reason is that it can be inefficient (and perhaps even impossible given the limited computational budget) to include every sampled points to construct a Gaussian process when i is large. This is especially true in our case since the decision space is continuous, and hence the number of sampled points will keep growing as the number of iterations grows.

Let $n_{i,j} = N_{V(i)-1}(\theta_{i,j})$; then we have taken $n_{i,j}$ observations for each $\theta_{i,j}$, $j = 1, \ldots, m(i)$, at the beginning of iteration i. Let $F_i(\theta_{i,j}) = \max\{f_{n_{i,j}}(\theta_{i,j}), \underline{M}\}$, and define $\bar{\mathbb{F}}_i = (F_i(\theta_{i,1}), \ldots, F_i(\theta_{i,m(i)}))^T$.

We build a response surface as follows: for every $\theta \in \Theta$, use

$$Y_i(\theta) = M(\theta) + \lambda_i(\theta)^T(\bar{\mathbb{F}}_i - \mathbb{M}_i) + \underline{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) < \xi_i\}} + \bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}} + \lambda_i(\theta)^T \mathcal{E}_i \qquad (66)$$

to model $f(\theta)$ in iteration $V(i)$, where $M(\theta)$ is defined in Section 5.2.1, $\underline{d}_i(\theta) = \min\{||\theta - \theta_{i,1}||, \ldots, ||\theta - \theta_{i,m(i)}||\}$, $\lambda_i(\theta) = (\lambda_{i,1}(\theta), \ldots, \lambda_{i,m(i)}(\theta))^T$ is a vector of weight functions, $\mathbb{M}_i = (M(\theta_{i,1}), \ldots, M(\theta_{i,m(i)}))^T$ is a vector of $M(\theta)$ evaluated at $\theta_{i,1}, \ldots, \theta_{i,m(i)}$, $\underline{\Delta}_i$ and $\bar{\Delta}_i$ are two normal random variables with the same mean 0 and variances $\underline{\sigma}_i^2$ and $\bar{\sigma}_i^2$, respectively, where $\underline{\sigma}_i > 0$ and $\bar{\sigma}_i \geq 0$, and $\mathcal{E}_i =$

$(\epsilon(\theta_{i,1}), \ldots, \epsilon(\theta_{i,m(i)}))^T$ is an $m(i)$-dimensional random vector following a multivariate normal distribution with mean 0 and covariance matrix

$$\Sigma_{\mathcal{E}_i} = \mathbf{diag}\left\{\frac{\sigma(\theta_{i,1})^2}{n_{i,1}}, \ldots, \frac{\sigma(\theta_{i,m(i)})^2}{n_{i,m(i)}}\right\}. \tag{67}$$

We construct a sampling distribution in a similar way as Sun, Hong, and Hu [65], except for the fact that we are concerned with a continuous decision space and we use the response surface (66) instead of (64). In other words, suppose that at the beginning of iteration $V(i)$, our best solution found in the last sampling step $V(i-1)$ is $\hat{f}^*_{i-1}$. Then the sampling distribution in iteration $V(i)$ is:

$$g_i(\theta) = \frac{P^*(Y_i(\theta) > \hat{f}^*_{i-1})}{\int_\Theta P^*(Y_i(\theta) > \hat{f}^*_{i-1})d\theta}. \tag{68}$$

In (66) we include two terms $\underline{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta)<\xi_i\}}$ and $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta)>\eta_i\}}$ in $Y_i(\theta)$ in addition to the response surface (64) of Sun, Hong, and Hu [65]. Since our feasible region is continuous, if the $\underline{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta)<\xi_i\}}$ term is not included, then candidate points θ that are close to any of $\{\theta_{i,1}, \ldots, \theta_{i,m(i)}\}$ are sampled with a low probability. This makes it difficult to improve further upon good sampled solutions. On the other hand, for candidate points θ that are far away from all $\theta_{i,1}, \ldots, \theta_{i,m(i)}$, we add the term $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta)>\eta_i\}}$ to increase the variance at θ, and thus increase the chance of sampling θ. We use the sample variance of $n_{i,j}$ observations of $f(\theta_{i,j})$ to estimate $\sigma^2(\theta_{i,j})$. However, whereas Sun, Hong, and Hu [65] bound the sample variance below, we do not do that since we have the term $\underline{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta)<\xi_i\}}$ to prevent the sample variance from being too small. Furthermore, $M(\theta)$, $\bar{\mathbb{F}}_i$, $\underline{\Delta}_i$, $\bar{\Delta}_i$, and $\mathcal{E}_i$ are mutually independent of each other.

Similar to Sun, Hong, and Hu [65], we need to choose the weight functions $\lambda_i(\theta)$ in a way to ensure the model will produce a desired sampling distribution in each sampling step. Next we discuss how to choose $\lambda_i(\theta)$: for any $\theta \in \Theta$ and $i \in \mathbb{N}^+$, $\lambda_i(\theta)$ satisfies:

(i) $\lambda_{i,j}(\theta) \geq 0$ for any $j = 1, \ldots, m(i)$;

(ii) $\sum_{j=1}^{m(i)} \lambda_{i,j}(\theta) = 1$;

(iii) $\lambda_{i,j}(\theta_k) = \mathbb{1}_{\{\theta_j = \theta_k\}}$, for all $j, k = 1, \ldots, m(i)$.

However, since we solve simulation optimization problems on continuous spaces, for any θ_j, $j = 1, \ldots, m(i)$, there may be $\theta \in \Theta$ that are arbitrarily close to θ_j (which is not the case in Sun, Hong, and Hu [65]). In order to define $\lambda_i(\theta)$ in a practical way, we let $u(\cdot)$ be a strictly decreasing function on $\mathbb{R}^+$, such that $u(x) > 0$ for any $x > 0$, $\lim_{x \to +\infty} u(x) = 0$, and $\lim_{x \to 0} u(x) = +\infty$ (this ensures we put more weight on the objective function estimates at points that are close to the sampled point). Let T^M be a large positive real number and T^m be a small non-negative real number. For each $\theta, \theta' \in \Theta$, and $\theta \neq \theta'$, let $\mathcal{K}(\theta, \theta') = \max\{\min\{T^M, u(||\theta - \theta'||)\}, T^m\}$. Then we can define:

$$\lambda_{i,j}(\theta) = \begin{cases} \frac{\mathcal{K}(\theta, \theta_{i,j})}{\sum_{k=1}^{m(i)} \mathcal{K}(\theta, \theta_{i,k})} & \text{if } \theta \notin \{\theta_{i,1}, \ldots, \theta_{i,m(i)}\}, \\ 1 & \text{if } \theta = \theta_{i,j}, \\ 0 & \text{if } \theta \in \{\theta_{i,1}, \ldots, \theta_{i,j-1}, \theta_{i,j+1}, \ldots, \theta_{i,m(i)}\}. \end{cases}$$

Due to the continuity of Θ, if a point is very close to (far away from) one or more previously visited points, the $u(\cdot)$ function can arbitrary close to ∞ (0). Hence, we introduce both T^M and T^m to prevent the cases $\frac{\infty}{\infty}$ and $\frac{0}{0}$ in any of the $\lambda_{i,j}(\theta)$. (Notice in Sun, Hong, Hu [65], since the feasible region is finite, T^M and T^m are not needed.)

Although the weight functions $\lambda_i(\theta)$ are not continuous in θ (unlike Sun, Hong, and Hu [65]), if we use $g_i(\cdot)$ as a sampling distribution, since the feasible region Θ is continuous, the probability of sampling any previously visited point is zero. Hence the probability that we hit discontinuity points of $\lambda_i(\theta)$ is actually zero.

5.3 Gaussian Search with Resampling and Discarding

In this section, we propose a new algorithm for solving continuous simulation optimization problems that incorporates the sampling distribution constructed in Section 5.2.2.

Let $\{K(i)\}_{i=1}^{\infty}$ be a nondecreasing sequence of positive integers. Let Θ_i^c denote the set of sampled, accepted, and not discarded ("current") solutions by the end of iteration $V(i)$. Let Θ_i^d denote the set of sampled, rejected or discarded ("discarded") solutions by the end of iteration $V(i)$. Let Θ_i^{c+} be the set of solutions sampled, accepted, and not discarded prior to the discarding procedure in iteration $V(i)$. Let Θ_i^{d+} be the set of solutions sampled, but then rejected or discarded, prior to the discarding procedure in iteration $V(i)$. Then $\tilde{\Theta}_i = \Theta_i^c \cup \Theta_i^d$ is the set of sampled solutions by the end of iteration $V(i)$. Finally, let Θ_i denote the set of sampled and accepted solutions by the end of iteration $V(i)$. The pseudo-code for our GSRD algorithm is given in Algorithm 3. Note that in our GSRD algorithm, we sample one point in each sampling step $V(i)$. However, we can also sample any fixed number of points (as Sun, Hong, and Hu [65] do) and the convergence result will not be affected.

For each i, let Γ_i be an $m(i) \times m(i)$ matrix whose (j,k)th element is $\gamma(\theta_{i,j}, \theta_{i,k})$, and let $\gamma_i(\theta)$ be an $m(i)$-dimensional vector whose jth element is $\gamma(\theta, \theta_{i,j})$. Therefore, we have:

$$\Gamma_i = \left(\gamma_i(\theta_{i,1}), \ldots, \gamma_i(\theta_{i,m(i)})\right).$$

For any $\theta \in \Theta$, $i \in \mathbb{N}^+ \backslash \{1\}$, using Proposition 1 of Sun, Hong, and Hu [65], we obtain:

$$E^*(Y_i(\theta)) = \lambda_i(\theta)^T \bar{\mathbb{F}}_i, \tag{69}$$

$$\begin{aligned} Var^*(Y_i(\theta)) =& Var^*\left(M(\theta) - \lambda_i(\theta)^T \mathbb{M}_i\right) + Var^*\left(\underline{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) < \xi_i\}}\right) \\ &+ Var^*\left(\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}\right) + Var^*\left(\lambda_i(\theta)^T \mathcal{E}_i\right) \\ =& \sigma^2\left[1 - 2\lambda_i(\theta)^T \gamma_i(\theta) + \lambda_i(\theta)^T \Gamma_i \lambda_i(\theta)\right] + \underline{\sigma}_i^2 \mathbb{1}_{\{\underline{d}_i(\theta) < \xi_i\}} \\ &+ \bar{\sigma}_i^2 \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}} + \lambda_i(\theta)^T \Sigma_{\mathcal{E}_i} \lambda_i(\theta). \end{aligned} \tag{70}$$

Next, we describe how to sample from the sampling distribution $g_i(\cdot)$. For any region $\Theta' \subseteq \Theta$, define $vol(\Theta') = \int_{\theta \in \Theta'} 1 d\theta$ as the volume of Θ'. Since our feasible region Θ is compact, we have $vol(\Theta) < \infty$. For each $i \in \mathbb{N}^+ \backslash \{1\}$, $\theta \in \Theta$, we have

Algorithm 3 Gaussian Search with Resampling and Discarding (GSRD)

Select three sequences of positive integers $\{K(i)\}$, $\{V(i)\}$, and $\{m(i)\}$, five sequences of real numbers $\{\delta_i\}_{i=1}^{\infty}$, $\{\xi_i\}_{i=1}^{\infty}$, $\{\eta_i\}_{i=1}^{\infty}$, $\{\underline{\sigma}_i^2\}$, and $\{\bar{\sigma}_i^2\}$, functions $h(\cdot)$ and $u(\cdot)$, parameters $\underline{M}$, σ, T^M, and T^m, a modeling strategy for selecting points in $\tilde{\Theta}_i$, a resampling strategy, and an acceptance criterion. Let $\Theta_0^c = \emptyset$, $\Theta_0^d = \emptyset$, $i = 1$, and $k = 0$.
while Stopping criterion is not satisfied **do**
  Let $k = k + 1$
  if $k = V(i)$ **then**
    if $k = 1$ **then**
      Select θ_1 in Θ, let $\Theta_1^{c+} = \{\theta_1\}$, $\Theta_1^{d+} = \emptyset$, update $N_k(\theta_1)$ and $S_k(\theta_1)$, and let $\hat{f}_1^* = \hat{f}_1(\theta_1)$
    else
      Choose $m(i)$ sampled solutions using the modeling strategy, indexed by $\{\theta_{i,1}, \ldots, \theta_{i,m(i)}\}$, from $\tilde{\Theta}_{i-1} = \Theta_{i-1}^c \cup \Theta_{i-1}^d$, construct $Y_i(\theta)$ according to (66), and construct a sampling distribution:

$$g_i(\theta) = \frac{P^*(Y_i(\theta) > \hat{f}_{i-1}^*)}{\int_\Theta P^*(Y_i(\theta) > \hat{f}_{i-1}^*) d\theta}.$$

      Let $\Theta_i^{c+} = \Theta_{i-1}^c$ and $\Theta_i^{d+} = \Theta_{i-1}^d$
      Sample θ_i according to $g_i(\theta)$ from Θ (see Algorithm 4). Based on the acceptance criterion, decide whether to include θ_i in either Θ_i^{c+} (so that $\Theta_i^{c+} = \Theta_i^{c+} \cup \{\theta_i\}$), or Θ_i^{d+} (so that $\Theta_i^{d+} = \Theta_i^{d+} \cup \{\theta_i\}$), and update $N_k(\theta_i)$ and $S_k(\theta_i)$
    end if
    For each $\theta \in \Theta_i^{c+}$, if $N_k(\theta) < K(i)$, obtain $K(i) - N_k(\theta)$ additional observations of $f(\theta)$ and update $N_k(\theta)$ and $S_k(\theta)$ accordingly
    Select an estimate of the current best solution $\theta_i^* \in \arg\max_{\theta \in \Theta_i^{c+}} \hat{f}_k(\theta)$
    Let $\hat{f}_i^* = \hat{f}_k(\theta_i^*)$
    Let $\Theta_i^c = \Theta_i^{c+}$ and $\Theta_i^d = \Theta_i^{d+}$
    For each $\theta \in \Theta_i^c$, if $\hat{f}_i^* - \hat{f}_k(\theta) > \delta_i$, move θ from Θ_i^c to Θ_i^d, and update $\Theta_i^c = \Theta_i^c \backslash \{\theta\}$ and $\Theta_i^d = \Theta_i^d \cup \{\theta\}$
    Let $i = i + 1$
  else
    Sample a solution θ from Θ_{i-1}^c using the resampling strategy
    Obtain an estimate of $f(\theta)$ and update $N_k(\theta)$ and $S_k(\theta)$
  end if
end while
Return θ_{i-1}^* as an estimate of the optimal solution.

$E^*(Y_i(\theta)) \leq f^*_{i-1}$, see (69). Because $Y_i(\theta)$ follows a normal distribution, we know that $P^*(Y_i(\theta) > f^*_{i-1}) \leq \frac{1}{2}$. Therefore,

$$g_i(\theta) = \frac{P^*(Y_i(\theta) > \hat{f}^*_{i-1})}{\int_\Theta P^*(Y_i(\theta) > \hat{f}^*_{i-1})d\theta} \leq \frac{\frac{1}{2}vol(\Theta)}{\int_\Theta P^*(Y_i(\theta) > \hat{f}^*_{i-1})d\theta} \times \frac{1}{vol(\Theta)}.$$

Motivated by Sun, Hong, and Hu [65], we define $U(x) = \frac{1}{vol(\Theta)}$ for $x \in \Theta$, and let $K_i := \frac{1}{2}[\int_\Theta P^*(Y_i(\theta) > \hat{f}^*_{i-1})d\theta]^{-1}vol(\Theta)$. Notice that $U(\theta)$ is the probability density function of a uniform distribution defined on the set Θ. Therefore we have $g_i(\theta) \leq K_i \times U(\theta)$, where K_i is a constant with respect to $\theta \in \Theta$. Hence, we use a similar acceptance-rejection method as Sun, Hong, and Hu [65]. The following is the detailed algorithm.

Algorithm 4 Acceptance-Rejection

1: Generate a sample Z uniformly in Θ and U uniformly in $(0, 1)$, where Z and U are independent
2: **if** $U \leq 2P^*(Y_i(Z)) > \hat{f}^*_{i-1})$ **then**
3: accept and set $\theta = Z$
4: **else**
5: go to Step 1
6: **end if**

For each $i \in \mathbb{N}^+\backslash\{1\}$, given all information from the past, we can derive the mean $E^*(Y_i(\theta))$ from equation (69) and the variance $Var^*(Y_i(\theta))$ for equation (70). Since $Y_i(\theta)$ follows a normal distribution, we are able to calculate $P^*(Y_i(Z) > \hat{f}^*_{i-1})$. Hence, the acceptance rejection method can be used to derive the sampling distribution.

From the above constructions, we know that a key step of our population-based simulation optimization algorithm is to utilize the sampled points to construct a Gaussian process, use the constructed Gaussian process to derive a sampling distribution, and sample from that distribution. However, it is not an easy task to sample new points from the derived sampling distribution. One way to sample from sampling distribution is to use acceptance rejection method as described in Algorithm 4. However, the acceptance rate will become lower as the number of iterations grows.

To benefit from the derived sampling distribution, rather than spend excessive effort on sampling exactly from it, we combine it with the simple point-based sampling scheme described in Section 3.3.2 that balances local and global search. We refer to the resulting algorithm as *Gaussian Adaptive Search with Resampling and Discarding* (GASRD). Explicitly, we first set a threshold value τ, which is a positive integer. In each sampling step $V(i) > 1$, we run Step 1 of Algorithm 4 at most τ times. If we successfully find (accept) a point θ, we set it as our newly sampled point in step $V(i)$. Otherwise, with probability $p > 0$, we sample uniformly from the entire feasible set Θ, and with probability $1 - p$, we sample uniformly from a neighborhood $N(\theta^*_{i-1})$ of the current estimate of the optimal solution.

5.4 Convergence Analysis

In this section, we provide the assumptions that are used in the convergence results. (Note that Assumption 5.4.1 below agrees with Assumption 3.2.2, except for the step numbers of the respective algorithms.) Then we prove the GSRD algorithm is globally convergent almost surely.

Assumption 5.4.1. *The random elements used for estimating the objective function values (e.g., in steps 12 and 20 of GSRD) are independent of the random elements used in the execution of algorithmic decisions (e.g., in steps 6, 8, and 19 of GSRD).*

Lemma 5.4.1. *Suppose Assumptions 3.2.1 and 5.4.1 hold. Choose $K(i) = \Psi(i^c)$ for some $c > 0$. If $c(l-1) > 2$, then $P(\hat{f}^*_i > f^* + \epsilon, i.o.) = 0$.*

Proof. Fix $\epsilon > 0$. For each $i \in \mathbb{N}^+$, consider:

$$P(\hat{f}^*_i > f^* + \epsilon) = P\left(\bigcup_{\theta \in \Theta^c_i} \hat{f}_{V(i)}(\theta) > f^* + \epsilon\right). \tag{71}$$

As in Andradóttir and Prudius [13], suppose that if a sampled point is rejected or discarded, we still collect additional observations at this point to ensure that it has

enough observations collected at it (i.e., by the end of iteration $V(i)$ it has at least $K(i)$ observations). Although we collect additional observations at the points in $\tilde{\Theta}_i \backslash \Theta_i^c$, we do not use them for making decisions concerning the evolution of the algorithm. Thus collecting additional data at these points does not impact convergence, and in practice we would not collect this data.

Note that $|\tilde{\Theta}_i| = i$ with probability one for GSRD ($|\tilde{\Theta}_i| < i$ is only possible if the same point is sampled multiple times) and $f(\theta) \leq f^*$ for all $\theta \in \Theta$. Therefore, we have,

$$\begin{aligned}
(71) &\leq P\left(\bigcup_{\theta \in \tilde{\Theta}_i} \hat{f}_{V(i)}(\theta) > f^* + \epsilon\right) \\
&\leq \sum_{j=1}^{i} P\left(\hat{f}_{V(i)}(\theta_j) > f^* + \epsilon\right) \\
&\leq \sum_{j=1}^{i} P\left(|\hat{f}_{V(i)}(\theta_j) - f(\theta_j)| > \epsilon\right) \\
&\leq \frac{Const}{i^{c(l-1)-1}}, \qquad (72)
\end{aligned}$$

where "Const" denote some positive constant number, and the last inequality is obtained due to $(7) - (12)$ in Section 3.2.2.

Since $c(l-1) > 2$, we have $\sum_{i=1}^{\infty} P(\hat{f}_i^* > f^* + \epsilon) \leq \sum_{i=1}^{\infty} \frac{Const}{i^{c(l-1)-1}} < \infty$. According to the first Borel-Cantelli lemma, we know that $P(\hat{f}_i^* > f^* + \epsilon, i.o.) = 0$. □

Lemma 5.4.2. *Suppose the assumptions in Lemma 5.4.1 hold. For each $\epsilon \in \mathbb{R}^+$, define $\Theta_\epsilon = \{\theta \in \Theta | f(\theta) \geq f^* - \epsilon\}$, and assume $vol(\Theta_\epsilon) > 0$. Suppose that $\xi_i \geq \underline{\xi}$ and $\underline{\sigma}_i \geq \underline{\sigma}$ for each $i \in \mathbb{N}^+$, where $\underline{\xi}, \underline{\sigma} > 0$. Let A_i be the event that the sampled point (θ_i) in iteration $V(i)$ is in Θ_ϵ. Then we have $P(A_i, i.o.) = 1$.*

Proof. For each $\theta \in \Theta$ and each $i \in \mathbb{N}^+, i > 1$, according to (70), we have

$$Var^*(Y_i(\theta)) \geq \sigma^2 \left[1 - 2\lambda_i(\theta)^T \gamma_i(\theta) + \lambda_i(\theta)^T \Gamma_i \lambda_i(\theta)\right] + \underline{\sigma}_i^2 \mathbb{1}_{\{\underline{d}_i(\theta) < \xi_i\}}.$$

From Proposition 2 of Sun, Hong, and Hu [65], we have

$$\sigma^2 \left[1 - 2\lambda_i(\theta)^T \gamma_i(\theta) + \lambda_i(\theta)^T \Gamma_i \lambda_i(\theta)\right] \geq \sigma^2 [1 - h(\underline{d}_i(\theta))]^2.$$

Therefore,

$$Var^*(Y_i(\theta)) \geq \sigma^2 [1 - h(\underline{d}_i(\theta))]^2 + \underline{\sigma}_i^2 \mathbb{1}_{\{\underline{d}_i(\theta) < \xi_i\}} \geq \min\{\sigma^2 [1 - h(\underline{d}(\xi_i))]^2, \underline{\sigma}_i^2\},$$

where we have used the fact that $h(\cdot)$ is a decreasing function taking values in $[0, 1]$. Let $\tilde{\sigma}^2 = \min\{\sigma^2 [1 - h(\underline{d}(\underline{\xi}))]^2, \underline{\sigma}^2\}$. Then we have $Var(Y_i(\theta)) \geq \tilde{\sigma}^2$. From (69), the non-negativity of $\lambda_i(\theta)$, and the definition of $\bar{\mathbb{F}}_i$, we also have: $E^*(Y_i(\theta)) \geq \underline{M}$.

Next, we bound the probability $P(A_i)$ as follows,

$$\begin{aligned}
P^*(Y_i(\theta) > \hat{f}_{i-1}^*) &\geq P^*(Y_i(\theta) > f^* + \epsilon) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}} \\
&= P^* \left(\frac{Y_i(\theta) - E^*(Y_i(\theta))}{\sqrt{Var^*(Y_i(\theta))}} > \frac{f^* + \epsilon - E^*(Y_i(\theta))}{\sqrt{Var^*(Y_i(\theta))}} \right) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}} \\
&\geq P^* \left(\frac{Y_i(\theta) - E^*(Y_i(\theta))}{\sqrt{Var^*(Y_i(\theta))}} > \frac{|f^* + \epsilon - \underline{M}|}{\tilde{\sigma}} \right) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}} \\
&= \bar{\Phi} \left(\frac{|f^* + \epsilon - \underline{M}|}{\tilde{\sigma}} \right) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}} \qquad (73)
\end{aligned}$$

(recall that $\bar{\Phi}(\cdot) = 1 - \Phi(\cdot)$, where $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal distribution). Therefore, we have

$$\begin{aligned}
g_i(\theta) &= \frac{P^*(Y_i(\theta) > \hat{f}_{i-1}^*)}{\int_\Theta P^*(Y_i(\theta) > \hat{f}_{i-1}^*) d\theta} \\
&\geq \frac{\bar{\Phi} \left(\frac{|f^* + \epsilon - \underline{M}|}{\tilde{\sigma}} \right) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}}}{\int_\Theta 1 d\theta} \\
&= \frac{1}{vol(\Theta)} \bar{\Phi} \left(\frac{|f^* + \epsilon - \underline{M}|}{\tilde{\sigma}} \right) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}}. \qquad (74)
\end{aligned}$$

Hence,

$$\begin{aligned}
P^*(A_i) &\geq \int_{\Theta_\epsilon} \frac{1}{vol(\Theta)} \bar{\Phi} \left(\frac{|f^* + \epsilon - \underline{M}|}{\tilde{\sigma}} \right) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}} d\theta \\
&= \frac{vol(\Theta_\epsilon)}{vol(\Theta)} \bar{\Phi} \left(\frac{|f^* + \epsilon - \underline{M}|}{\tilde{\sigma}} \right) \mathbb{1}_{\{\hat{f}_{i-1}^* \leq f^* + \epsilon\}}. \qquad (75)
\end{aligned}$$

Since Θ is compact and we assume $vol(\Theta_\epsilon) > 0$, we have $0 < vol(\Theta_\epsilon) \leq vol(\Theta) < +\infty$. Moreover, $\bar{\Phi}(\frac{|f^*+\epsilon-\underline{M}|}{\tilde{\sigma}})$ is a positive constant. Therefore we have

$$\sum_{i=1}^{\infty} P^*(A_i) \geq \frac{vol(\Theta_\epsilon)}{vol(\Theta)} \bar{\Phi}\left(\frac{f^*+\epsilon-\underline{M}}{\tilde{\sigma}}\right) \sum_{i=1}^{\infty} \mathbb{1}_{\{\hat{f}^*_{i-1} \leq f^*+\epsilon\}}.$$

From Lemma 5.4.1, we have $P(\hat{f}^*_i \leq f^* + \epsilon, a.a.) = 1$. Hence $\sum_{i=1}^{\infty} \mathbb{1}_{\{\hat{f}^*_{i-1} \leq f^*+\epsilon\}} = \infty$ with probability one. Therefore, from Corollary 2.3 of Hall and Heyde [30] (the conditional Borel-Cantelli lemma), we know that $P(A_i, i.o.) = 1$. This completes the proof. □

We use the [AH] acceptance criterion described in Section 3.2.3. Explicitly: Let $\delta > 0$ and let $\{H(i)\}$ be a sequence of positive integers. At step $V(i)$, we obtain $H(i)$ independent observations of $f(\theta)$ after we sample a new point θ. The newly sampled solution θ is included in the set Θ_i^{c+} of sampled, accepted, and not discarded points in iteration i if an objective function estimate based on $H(i)$ observations at this point is at least as good as the estimated objective function value at the best solution found in the last sampling step (step $V(i-1)$) minus an indifference parameter δ. Explicitly, if $\hat{f}^*_{V(i-1)} - f_{H(i)}(\theta_i) \leq \delta$, then accept the sampled point θ and put it into Θ_i^{c+}, otherwise, reject this point.

Theorem 5.4.1. *Suppose the assumptions in Lemmas 5.4.1 and 5.4.2 hold. Choose $\delta_i = \Omega(i^{-\gamma})$ for each $i \in \mathbb{N}^+$, and assume that $c(l-1) - 2\gamma l > 2$. Use GSRD with acceptance criterion [AH], where $H(i) = \Omega(i^q)$ for all i, $q > 0$, and $ql > 1$. Then $f(\theta_i^*) \to f^*$ almost surely as $i \to \infty$.*

Proof. Fix any $0 < \epsilon < \delta$, and let $\bar{\Theta}_\epsilon = \Theta \backslash \Theta_\epsilon$. We need to show:

$$P\left(\theta_i^* \in \bar{\Theta}_\epsilon, i.o.\right) = 0.$$

From Lemma 5.4.2, we know we sample points in Θ_ϵ infinitely often. Since we use acceptance criterion [AH], from Proposition 3.2.3 in Section 3.2.3, it follows that we

have $P(\theta_i \in \Theta_i \cap \Theta_\epsilon, i.o.) = 1$. The result now follows directly from Theorem 3.2.1 in Section 3.2.2. □

5.5 Numerical Analysis

The main contribution of this chapter is to design a Gaussian sampling algorithm that adaptively uses the available information to solve continuous simulation optimization problems. In this section, to investigate the effects of utilizing the available population of sampled points, rather than only the single current estimated optimal point, we compare the Gaussian Adaptive Search with Resampling and Discarding (GASRD) algorithm developed in this chapter with the ASRD implementation in Chapter 3.

The outline of this section is as follows: In Section 5.5.1, we describe our test problems, in Section 5.5.2, we provide implementation details of the tested algorithms, and in Section 5.5.3, we compare and analyze our numerical results.

5.5.1 Test Problems

This section describes our test problems. The following four benchmark problems, which have been previously studied, e.g., in Andradóttir and Prudius [13] and Sun, Hong, and Hu [65], are used in our experiments. The first two are two-dimensional problems that have simple structures. The third is a two-dimensional problem with multiple local optima. The fourth is a five-dimensional, highly multimodal problem problem.

The Smooth problem (Andradóttir and Prudius [13]; Section 3.3.1):

$$f(\theta) = -[(x_1 - 0.5)\sin(10x_1) + (x_2 + 0.5)\cos(5x_2)],$$

$\Theta = \{(x_1, x_2) \subseteq \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 1\}$, and for each $\theta \in \Theta$, $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ and $X(\omega)$ is a $\mathcal{N}(0, 1)$ random variable. The approximate range of the objective function values is $(-3, 1.502]$. The optimal value is $f^* \simeq 1.502$.

The Two Hills problem (Andradóttir and Prudius [13]; Section 3.3.1 with different noise):

$$f(\theta) = \max\{f_1(\theta), f_2(\theta), 0\},$$

where $f_1(\theta) = -(0.4x_1 - 5)^2 - 2(0.4x_2 - 17.2)^2 + 7$ and $f_2(\theta) = -(0.4x_1 - 12)^2 - (0.4x_2 - 4)^2 + 4$. The feasible region is given by $\Theta = \{(x_1, x_2) \in \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 50\}$. We let $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ for all $\theta \in \Theta$, as for the smooth problem, with $X(\omega)$ being $\mathcal{N}(0, 10)$ for all $\theta \in \Theta$. This objective function is of interest, mainly, because it has two hills of different heights (4 and 7), located relatively far apart (the hill of height 4 is centered at $(30, 10)$ and the hill of height 7 is centered at $(12.5, 43)$), and separated by a flat valley (of height 0). The range of the objective function values is $[0, 7]$. The optimal value is $f^* = 7$.

The Multiple Local Optima problem (Sun, Hong, and Hu [65]):

$$f(\theta) = 10\frac{\sin^6(0.05\pi x_1)}{2^{2(\frac{x_1-90}{50})^2}} + 10\frac{\sin^6(0.05\pi x_2)}{2^{2(\frac{x_2-90}{50})^2}}. \tag{76}$$

The feasible region is $\Theta = \{(x_1, x_2) \in \mathbb{R}^2 : 0 \leq x_1, x_2 \leq 100\}$. The function f has 25 local optima with the global optimum $(90, 90)$ satisfying $f(90, 90) = 20 = f^*$. We let $h(\theta, X(\omega)) = f(\theta) + X(\omega)$ for all $\theta \in \Theta$, as for the above two problems, with $X(\omega)$ being $\mathcal{N}(0, 10)$ for all $\theta \in \Theta$.

The Pinter 5D problem (Hu, Fu, and Marcus [37]; Section 3.3.1 with different dimension):

$$\begin{aligned} f(\theta) = &- \left(\sum_{i=1}^{s} i x_i^2 + \sum_{i=1}^{s} i \sin^2 \left(x_{i-1} \sin x_i - x_i + \sin x_{i+1} \right) \right) \\ &- \left(\sum_{i=1}^{s} i \log_{10} \left[1 + i(x_{i-1}^2 - 2x_i + 3x_{i+1} - \cos x_i + 1)^2 \right] \right) - 1, \end{aligned}$$

where $x_0 = x_s$, $x_{s+1} = x_1$, and $s = 5$. The feasible region is $\Theta = \{(x_1, \ldots, x_s) \in \mathbb{R}^s : -10 \leq x_i \leq 10, i = 1, \ldots, s\}$. The form of $h(\theta, X(\omega))$ is as for the other three

Table 7: Notation

Notation	Algorithm
ASRD	ASRD with $p = 0.5$ and acceptance criterion [AH]
RSRD	ASRD with $p = 1$ and acceptance criterion [AH]
GASRD	GASRD with $p = 0.5$
GRSRD	GASRD with $p = 1$
GASRD_0	GASRD with $p = 0.5$ and $\bar{\sigma}_i = 0$
GRSRD_0	GASRD with $p = 1$ and $\bar{\sigma}_i = 0$

test problems, with $X(\omega)$ being $\mathcal{N}(0, 100)$ for all $\theta \in \Theta$. The approximate range is $(-1100, -1]$, and this problem is highly multimodal with a global maximum at $(0, \ldots, 0)$ and $f^* = -1$.

5.5.2 Algorithm Implementation

In this section, we will provide details for our GASRD and ASRD algorithms. Since the term $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}$ in (66) does not affect the convergence result, we will test GASRD framework without this term ($\bar{\sigma}_i = 0$). Finally, under both GASRD and ASRD frameworks, for different values of p, we give different names to the corresponding frameworks as listed in Table 7, and we compare the performance of each algorithm in Table 7.

As in Section 3.3.2, define:

$$N(\theta) = N((x_1, \ldots, x_s)) = \{(x'_1, \ldots, x'_s) \in \Theta : |x_i - x'_i| \leq r, i = 1, \ldots, s\}$$

for all $\theta \in \Theta$. Here we use $r > 0$ to denote the radius of the "local" neighborhood.

For the resampling strategy of GASRD, we use the same as that of ASRD in Chapter 3. The implementation details are as follows: Let $V(i) = \lfloor i^v \rfloor$, where $v \geq 1$, and note that $s_k = \lfloor k^{1/v} \rfloor$ is the number of points sampled by the end of iteration k. Then, a point $\theta \in \Theta^c_{s_k}$ is resampled in iteration k with probability

$$p_k(\theta) = \frac{\exp\{\hat{F}_{s_k}(\theta)\}}{\sum_{\theta' \in \Theta^c_{s_k}} \exp\{\hat{F}_{s_k}(\theta')\}},$$

where $\hat{F}_{s_k}(\theta) = \min\{\max\{\underline{U}, \hat{f}_{s_k}(\theta)/T\}, \overline{U}\}$, with $\overline{U} > \underline{U} > 0$ and $T > 0$. Here we choose $\overline{U} = 400$, $\underline{U} = -400$, $v = 1.1$, and $T = 0.1$.

Finally, we discuss how to choose parameter values. For the ASRD algorithm, choose $\delta_i = D/i^\gamma$, where $\gamma \geq 0$. Let $\gamma = 0.2$ for all test problems. Choose $D = 1$ for the Smooth problem; $D = \sqrt{10}$ for the Two Hills and Multiple Local Optima problems; and $D = 10$ for Pinter 5D problem. Here we choose D to be the standard deviation of the noise in objective function (in practice, these values would need to be estimated). Let $K(i) = \lceil Si^c \rceil$ and $H(i) = \lceil Qi^q \rceil$, where $S, c, Q, q > 0$, and choose $S = 1$, $c = 0.5$, $Q = 1$, and $q = 0.05$. Since the noise follows a normal distribution, which corresponds to $l = \infty$; for all of the test problems, we only need $c > 2\gamma$ and $q > 0$. Moreover, choose $r = 0.01$ for the Smooth problem, $r = 0.5$ for the Two Hills problem, $r = 1$ for the Multiple Local Optima problem, and $r = 0.2$ the Pinter 5D problem. Here r is chosen to be $\frac{1}{100}$ of the diameter of the feasible region for each test problem. The additional parameter for the acceptance criterion is $\delta = 0.1$. We take five objective function observation in each re-sampling step. Most of the parameter values are chosen as in Andradóttir and Prudius [13] and Chapter 3; how to determine a priori the most appropriate values of all the parameters is outside of the scope of this thesis.

For GASRD, let $\tau = 10$, $\gamma(\theta, \theta') = \exp\{-||\theta - \theta'||^{0.5}\}$, $u(x) = x^{-4}$, $T^M = 10^5$, $T^m = 10^{-6}$, and $\underline{M} = -10^{10}$. In step 8 of Algorithm 3 (when $i > 1$), we chose the points from Θ^c_{i-1} and Θ^d_{i-1} as follows. Select $\{m_c(i)\}$ and $\{m_d(i)\}$, two sequences of positive integers. If $|\Theta^c_i| < m_c(i)$ ($|\Theta^d_i| < m_d(i)$), choose all the points in Θ^c_i (Θ^d_i) to be in the construction of $Y_i(\cdot)$; otherwise, choose $m_c(i)$ ($m_d(i)$) points randomly from Θ^c_i (Θ^d_i). Hence, $m(i) = \min\{m_c(i), |\Theta^c_i|\} + \min\{m_d(i), |\Theta^d_i|\}$. The reason why we choose a certain number of points from both Θ^c_i and Θ^d_i is because we would like to put a higher probability around good points (by choosing points from Θ^c_i) and simultaneously avoid regions with inferior points (by choosing points from Θ^d_i).

Therefore, we choose $m_c(i) = m_d(i) = 10$ for the Smooth, Two Hills, and Multiple Local Optima problems, and choose $m_c(i) = m_d(i) = 20$ for the Pinter 5D problem.

According to the numerical analysis by Sun, Hong, and Hu [65], σ^2 (the variance of the unconditional Gaussian process $M(\theta)$) balances the exploitation and exploration trade-off. Explicitly, when σ^2 is large, the Gaussian search spends more efforts in exploration, whereas when σ^2 is small, the Gaussian search spends more efforts in exploitation. As exploration is much less efficient in higher dimensions, we emphasize exploitation for large s, by using a smaller value for σ^2. Hence, we choose σ, to be twice as much as the standard deviation of the noise in objective function for the Smooth, Two Hills, and Multiple Local Optima problems, and we choose σ^2 to be half as much as the standard deviation of the noise in objective function for the Pinter 5D problem (again, in practice, these values would need to be estimated). Also, for each $i > 1$, choose $\bar{\sigma}_i$, and $\underline{\sigma}_i$ to be the same as standard deviation of the noise in objective function respectively (except for GASRD$_0$ and GRSRD$_0$). Hence $\sigma = 2$, $\bar{\sigma}_i = 1$, and $\underline{\sigma}_i = 1$ for the smooth problem, $\sigma = 2\sqrt{10}$, $\bar{\sigma}_i = \sqrt{10}$, and $\underline{\sigma}_i = \sqrt{10}$ for the Two Hills and Multiple Local Optima problems, $\sigma = 5$, $\bar{\sigma}_i = 10$, and $\underline{\sigma}_i = 10$ for the Two Hills and Multiple Local Optima problems. Lastly, let ξ_i and η_i be 1% and 10% of the diameter of the feasible region of each problem respectively. Therefore, we have $\xi_i = 0.01$ and $\eta_i = 0.1$ for the smooth problem, $\xi_i = 0.5$ and $\eta_i = 5$ for the Two Hills problem, $\xi_i = 1$ and $\eta_i = 10$ for the Multiple Local Optima problem, and $\xi_i = 0.2$ and $\eta_i = 2$ for the Pinter 5D problem.

5.5.3 Performance Comparison

Figures 23, 24, 25, and 26 show the empirical performance of the GASRD, GASRD$_0$, ASRD, GRSRD, GRSRD$_0$, and RSRD methods on the Smooth, Two Hills, Multiple Local Optima, and Pinter 5D problems, respectively. Explicitly, on the left side of each figure, we compare Gaussian search and point-based adaptive search, and hence

document the performance of ASRD, GASRD, RSRD, and GRSRD. By contrast, on the right side of each figure, we investigate the effects of the term $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}$ on Gaussian search, and hence document the performance of GASRD, GASRD$_0$, GRSRD, and GRSRD$_0$. Let $N_k = \sum_{\theta \in \tilde{\Theta}_{s_k}} N_k(\theta)$ be the total number of objective function evaluations by the end of iteration k. Let N be the simulation budget. The performance of the algorithms is averaged over 100 independent replications for all four test problems. Their performance is documented by plotting 100 pairs (x, y), where $x \in \{0.01N, 0.02N, \ldots, N\}$, and y is the average objective function value at the estimated optimal solution after x objective function observations have been collected. As the estimate of the optimal solution is only updated in iterations $V(1)$, $V(2)$,..., the value of y is the same for all corresponding $x \in [N_{V(i)}, N_{V(i+1)})$. We plot $-f(\theta^*_{m_k})$ for the Pinter 5D problem (as it has negative objective function values), therefore smaller values are better for Figure 26 (larger values are better for Figures 23, 24, and 25). The sequence in which the numerical results are presented moves from lower dimensional, smoother problems to higher dimensional problems with greater curvature.

We emphasize that:

- The performances documented in the figures do not take into account overhead of each algorithm;
- The results are good approximations when objective function evaluations are very expensive.

In Table 8, we provide the average CPU running time (in seconds) over the 100 independent replications of each algorithm on each test problem. In the following, we will analyze these four problems in detail.

For the Smooth problem, we can see from the left side of Figure 23 that ASRD performs the best among all algorithms with $\bar{\sigma} > 0$, and the other three algorithms

Table 8: Average CPU time needed (in seconds) to run the algorithm

	ASRD	RSRD	GASRD	GRSRD	GASRD_0	GRSRD_0
Smooth	0.278	0.295	1.614	1.668	1.804	1.872
Two Hills	0.383	0.440	3.316	3.344	3.453	3.380
Multiple Local Optima	0.403	0.398	5.811	5.667	5.889	5.597
Pinter 5D	0.465	0.537	20.764	20.316	20.483	20.398

have similar performance. The main reason is that the smooth problem is unimodal, ASRD has balanced local and global search steps and is able to identify the optimal solution more accurately by conducting local exploitation (in fact, global exploration is unnecessary for this test problem). Moreover, we notice from the right side of Figure 23 that the term $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}$ does not significantly affect performance. On the other hand, from Table 8, we can see that on average, the GSRD framework takes more than five times longer than ASRD to run.

For the Two Hills problem, from the left side of Figure 24, we notice that GASRD performs better than ASRD, and simultaneously GRSRD performs better than RSRS. Moreover, ASRD has better performance that RSRD after 2000 objective function evaluations, whereas GASRD and GRSRD have similar performance. We can see that in this case, the Gaussian sampling strategy (which utilizes more past information than both pure random search as well as balanced local and global search) achieves better performance. From the right side of Figure 24, we can see that similar to the smooth problem, the term, $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}$ does not significantly affect performance. However, from Table 8, for the Two Hills problem, the GSRD framework is more than eight times slower than ASRD.

For the Multiple Local Optima problem, from the left side of Figure 25, it is evident that GASRD has much better performance than ASRD, simultaneously, GRSRD has much better performance that RSRD. Moreover, ASRD performs better than RSRD at early stages of the simulation and both have similar performance at later stages. Similarly, GASRD has better performance than GRSRD at early stages of the

simulation but have similar performance later on as the number of iterations grows. Here, since the test problem has several local optimal solutions, we can see from the numerical results that applying the Gaussian sampling strategy (that utilizes the information on multiple previously sampled points) can greatly improve performance. Moreover, from the right side of Figure 25, we notice that GASRD_0 performs better than GASRD, and GRSRD_0 performs better than GRSRD. In this case, unlike the Smooth and Two Hills problems, it is better not to include the term $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}$ in the Gaussian process. Finally, for the running time, we can see from Table 8 that the GSRD framework is at least twelve times slower than the ASRD framework.

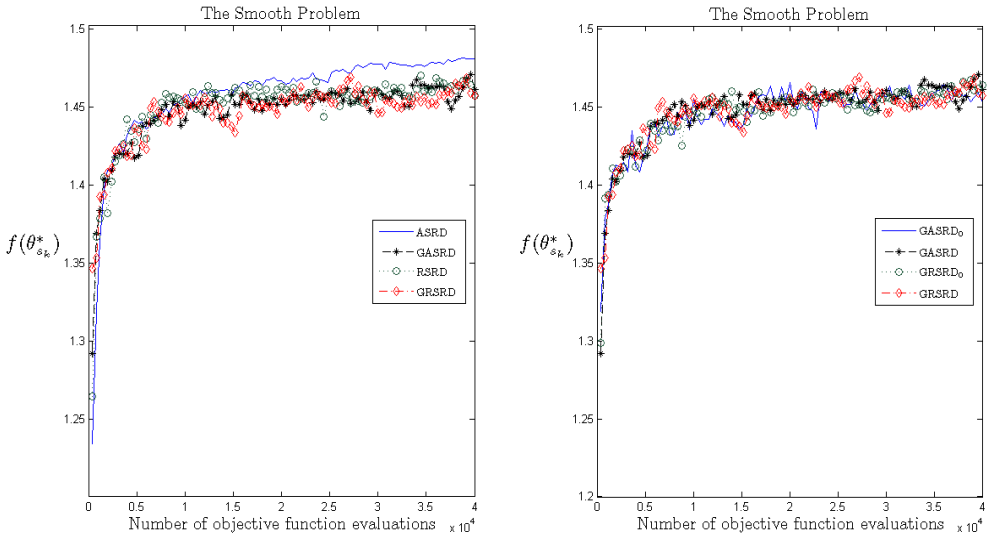

Figure 23: Approximate performance of the algorithms on the Smooth problem when objection function observations are expensive

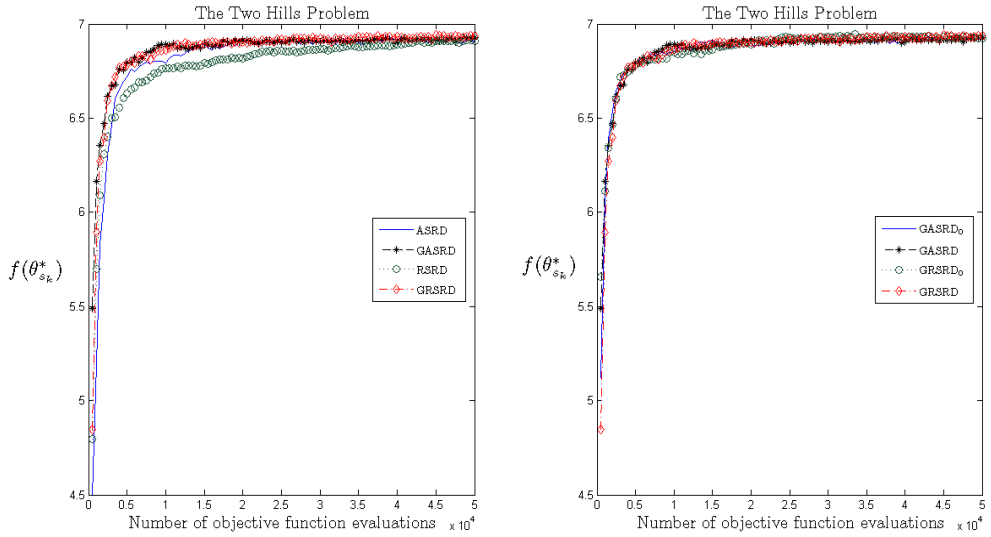

Figure 24: Approximate performance of the algorithms on the Two Hills problem when objection function observations are expensive

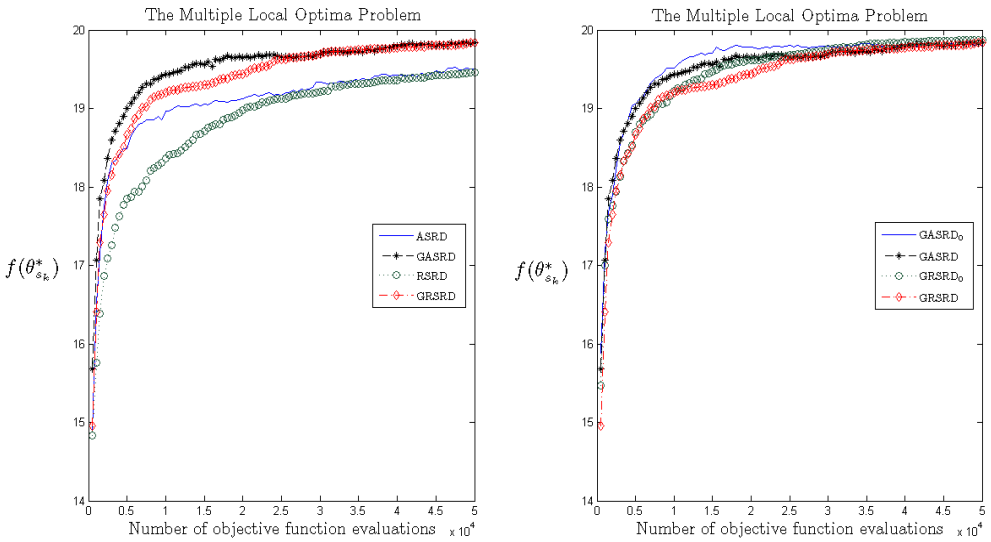

Figure 25: Approximate performance of the algorithms on the Multiple Local Optima problem when objection function observations are expensive

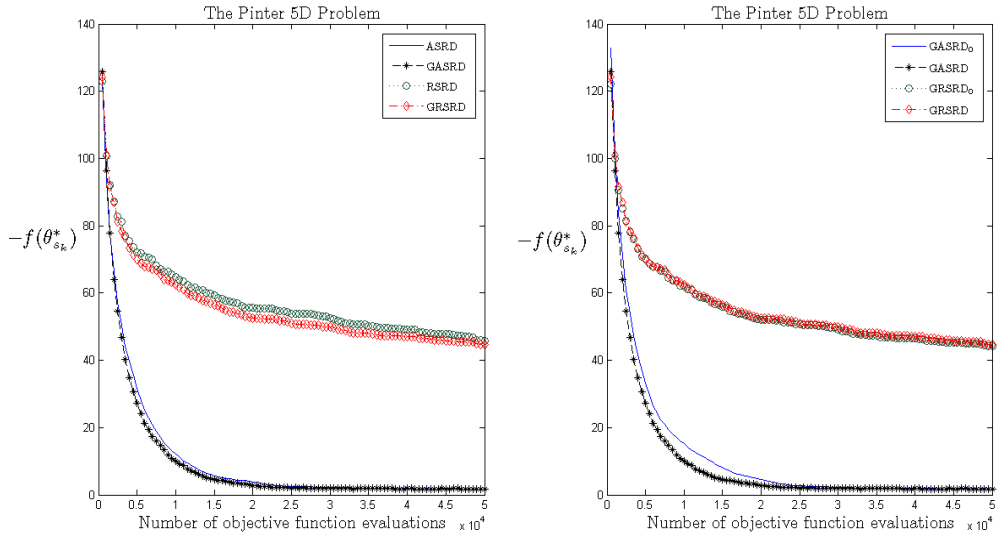

Figure 26: Approximate performance of the algorithms on the Pinter 5D problem when objection function observations are expensive

For the Pinter 5D problem, numerical results on the left side of Figure 26 show that GASRD and ASRD perform significantly better than GRSRD and RSRD. Moreover, GRSRD performs noticeably better than RSRD, whereas GASRD performs slightly better than ASRD at early stages of the simulation and both have similar performance at later stages. On the other hand, from the right side of Figure 26, GASRD performs noticeably better than GASRD$_0$ whereas GRSRD$_0$ and GRSRD have similar performance. We conclude that when the dimension of the problem gets higher, adaptive search is advantageous over pure random search. Also, including the term $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}$ can actually help the GSRD framework perform better. But, from Table 8, we notice that here Gaussian search is about 40 times slower than adaptive search.

In summary, from Figures 23 through 26 and Table 8, we observe:

- GASRD takes much longer to run than ASRD, especially on higher dimensional problems;
- adaptive search has better performance than pure random search;

- when overhead is not taken into account, GASRD has better performance over ASRD in most cases, except for the smooth problem which is unimodal;
- whether to include the term $\bar{\Delta}_i \mathbb{1}_{\{\underline{d}_i(\theta) > \eta_i\}}$ into Gaussian process or not depends on the tested problems.

5.6 Conclusion

In this chapter we develop a Gaussian Search with Resampling and Discarding (GSRD) algorithm by incorporating a Gaussian sampling distribution in the Adaptive Search with Resampling and Discarding (ASRD) framework of Chapter 3. We prove that GSRD converges almost surely and provide numerical analysis showing that when the objective function is multi-modal, objective function observations are expensive, and the underlying problem dimension is not high, the GSRD framework makes improvements over a point-based implementation of ASRD, and is fairly robust (never performs poorly).

CHAPTER VI

CONTRIBUTIONS AND FUTURE RESEARCH

This thesis investigates how to design provably convergent algorithms to solve simulation optimization problems. Such problems are generally hard to solve due to uncertainties involved in the problem formulation and lack of structure that could be utilized by traditional optimization techniques. In my thesis, I use adaptive random search to build three convergent frameworks, ASRD, ASDP, GSRD, to solve two different classes of simulation optimization problems, namely continuous optimization problems with stochastic objective functions, and with both stochastic objective functions and stochastic constraints.

The main contributions of my thesis are:

1. The frameworks guarantee almost surely convergence, which is valuable not only in academic research but also to practitioners in that it ensures that additional effort will lead to improved solutions, and that with sufficient effort, the algorithms will return solutions that are arbitrarily close to optimal;

2. ASDP is the first simulation optimization algorithm that converges from inside the feasible region as far as we know;

3. ASRD and ASDP frameworks are generic, in that different versions can be applied to a wide variety of problems;

4. Numerical results show that our frameworks are promising.

With regards to our work, there are several future research directions that could be pursued:

1. In Chapter 3, one important line of future research is to develop efficient criteria to balance the simulation budget between sampling and resampling.

2. In Chapter 4, an interesting future research line is to utilize the sampled points to estimate the feasible region and subsequently to guide the sampling distribution towards the feasible region.

3. In Chapter 5, numerical analysis shows that Gaussian sampling is slow, especially in high dimensions. The development of fast sampling distributions that can automatically balance exploitation and exploration is a worthwhile direction of future research.

REFERENCES

[1] Alrefaei, M. H. and Andradóttir, S., "A simulated annealing algorithm with constant temperature for discrete stochastic optimization," *Management Science*, vol. 45, pp. 748–764, May 1999.

[2] Alrefaei, M. H. and Andradóttir, S., "A modification of the stochastic ruler method for discrete stochastic optimization," *European Journal of Operational Research*, vol. 133, no. 1, pp. 160–182, 2001.

[3] Alrefaei, M. and Andradóttir, S., "Discrete stochastic optimization using variants of the stochastic ruler method," *Naval Research Logistics*, vol. 52, pp. 344–360, June 2005.

[4] Andradóttir, S., "A method for discrete stochastic optimization," *Management Science*, vol. 41, pp. 1946–1961, December 1995.

[5] Andradóttir, S., "A stochastic approximation algorithm with varying bounds," *Operations Research*, vol. 43, pp. 1037–1048, 1995.

[6] Andradóttir, S., "A global search method for discrete stochastic optimization," *SIAM Journal on Optimization*, vol. 6, pp. 513–530, 1996.

[7] Andradóttir, S., "A review of simulation optimization techniques," *Proceedings of the 1998 Winter Simulation Conference*, pp. 151–158, 1998.

[8] Andradóttir, S., "Accelerating the convergence of random search methods for discrete stochastic optimization," *ACM Transactions on Modeling and Computer Simulation*, vol. 9, pp. 349–380, October 1999.

[9] Andradóttir, S., "Simulation optimization with countably infinite feasible regions: Efficiency and convergence," *ACM Transactions on Modeling and Computer Simulation*, vol. 16, pp. 357–374, October 2006.

[10] Andradóttir, S., "An overview of simulation optimization via random search, Chapter 20 in *Handbooks in Operations Research and Management Science: Simulation*," edited by S. G. Henderson and B. L. Nelson. Elsevier Science, Amsterdam, 2006.

[11] Andradóttir, S. and Kim, S., "Fully sequential procedures for comparing constrained systems via simulation," *Naval Research Logistics*, vol. 57, pp. 403–421, August 2010.

[12] Andradóttir, S. and Prudius, A. A., "Balanced explorative and exploitative search with estimation for simulation optimization," *INFORMS Journal on Computing*, vol. 21, pp. 193–208, September 2009.

[13] Andradóttir, S. and Prudius, A. A., "Adaptive random search for continuous simulation optimization," *Naval Research Logistics*, vol. 57, pp. 583–604, September 2010.

[14] Andradóttir, S. and Prudius, A. A., "Averaging frameworks for simulation optimization with applications to simulated annealing," *Naval Research Logistics*, vol. 59, pp. 411–429, September 2012.

[15] Baumert, S. and Smith, R. L., "Pure random search for noisy objective functions," *Technical Report 01-03, University of Michigan*, May 2002.

[16] Birbil, S., Fang, S., and Sheu, R., "On the convergence of a population-based global optimization algorithm," *Journal of Global Optimization*, vol. 30, pp. 301–318, 2004.

[17] Carson, Y. and Maria, A., "Simulation optimization: Methods and applications," *Proceedings of the 1997 Winter Simulation Conference*, pp. 118–126, 1997.

[18] Dentcheva, D. and Ruszczynski, A., "Optimization with stochastic dominance constraints," *SIAM Journal on Optimization*, vol. 14, pp. 548–566, 2003.

[19] Fox, B. and Heine, G., "Probabilistic search with overrides," *The Annals of Applied Probability*, vol. 5, pp. 1087–1094, 1995.

[20] Fu, M. C., "Gradient estimation, Chapter 19 in *Handbooks in Operations Research and Management Science: Simulation*," edited by S. G. Henderson and B. L. Nelson. Elsevier Science, Amsterdam, 2006.

[21] Fu, M., "Optimization for simulation: Theory vs. practice," *INFORMS Journal on Computing*, vol. 14, pp. 192–215, 2002.

[22] Fu, M., Glover, F., and April, J., "Simulation optimization: methods and applications," *Proceedings of the 2005 Winter Simulation Conference*, pp. 83–95, 2005.

[23] Gelfand, S. and Mitter, S., "Simulated annealing with noisy or imprecise energy measurement," *Journal of Optimization Theory and Applications*, vol. 62, pp. 49–62, 1989.

[24] Goldsman, D. and Nelson, B., "Comparing systems via simulation, Chapter 8 in *Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice*," edited by J. Banks. Wiley, New York, 1998.

[25] Gong, W., Ho, Y., and Zhai, W., "Stochastic comparison algorithm for discrete optimization with estimation," *Proceedings of the 31st IEEE Conference on Decision and Control*, pp. 795–800, 1992.

[26] Gong, W., Ho, Y., and Zhai, W., "Stochastic comparison algorithm for discrete optimization with estimation," *SIAM Journal on Optimization*, vol. 10, no. 2, pp. 384–404, 2000.

[27] Gutjahr, W. and Pflug, G., "Simulated annealing for noisy cost functions," *Journal of Global Optimization*, vol. 8, pp. 1–13, 1996.

[28] Haddock, J. and Mittenthal, J., "Simulation optimization using simulated annealing," *Computers & Industrial Engineering*, vol. 22, pp. 387–395, 1992.

[29] Hajek, B. and Sasaki, G., "Simulated annealing – to cool or not," *Systems & Control Letters*, vol. 12, pp. 443–447, 1989.

[30] Hall, P. and Heyde, C., *Martingale Limit Theory and Its Application*. Academic Press, New York, NY, 1980.

[31] Healey, C., Andradóttir, S., and Kim, S., "Efficient comparison of constrained systems using dormancy," *European Journal of Operational Research*, vol. 224, pp. 340–352, 2013.

[32] Healy, K. and Schruben, L., "Retrospective simulation response optimization," *Proceedings of the 1991 Winter Simulation Conference*, pp. 901–906, 1991.

[33] Hong, J. and Nelson, B., "A brief introduction to optimization via simulation," *Proceedings of the 2009 Winter Simulation Conference*, pp. 75–85, 2009.

[34] Hong, L. and Nelson, B., "Discrete optimization via simulation using COMPASS," *Operations Research*, vol. 54, no. 1, pp. 283–298, 2006.

[35] Hong, L. and Nelson, B., "A sequential procedure for neighborhood selection-of-the-best in optimization via simulation," *European Journal of Operational Research*, vol. 173, pp. 283–298, 2006.

[36] Hong, L. and Nelson, B., "A framework for locally convergent random-search algorithms for discrete optimization via simulation," *ACM Transactions on Modeling and Computer Simulation*, vol. 17, no. 4, p. Article 19, 2007.

[37] Hu, J., Fu, M. C., and Marcus, S. I., "A model reference adaptive search method for global optimization," *Operations Research*, vol. 55, no. 3, pp. 549–568, 2007.

[38] Hu, J., Fu, M. C., and Marcus, S. I., "A model reference adaptive search method for stochastic global optimization," *Communications in Information and Systems*, vol. 8, no. 3, pp. 245–276, 2008.

[39] Hu, J. and Hu, P., "Annealing adaptive search, cross-entropy, and stochastic approximation in global optimization," *Naval Research Logistics*, vol. 58, pp. 457–477, August 2011.

[40] Hu, L. and Andradóttir, S., "A penalty function approach for simulation optimization with stochastic constraints," *Proceedings of the 2014 Winter Simulation Conference*, pp. 3730–3736, 2014.

[41] Huang, D., Allen, T., Notz, W., and Zeng, N., "Global optimization of stochastic black-box systems via sequential kriging meta-models," *Journal of Global Optimization*, vol. 34, pp. 441–466, 2006.

[42] Hunter, S. and Pasupathy, R., "Optimal sampling laws for stochastically constrained simulation optimization on finite sets," *INFORMS Journal on Computing*, vol. 25, pp. 527–542, summer 2013.

[43] Jones, D., "A taxonomy of global optimization methods based on response surfaces," *Journal of Global Optimization*, vol. 21, pp. 345–383, 2001.

[44] Kabirian, A. and Ólafsson, S., "Selection of the best with stochastic constraints," *Proceedings of the 2009 Winter Simulation Conference*, pp. 574–583, 2009.

[45] Kiefer, J. and Wolfowitz, J., "Stochastic estimation of the maximum of a regression function," *Annals of Mathematical Statistics*, vol. 23, pp. 462–466, 1952.

[46] Kirkpatrick, S., Gelatt, C., and Vecchi, M., "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.

[47] Kleywegt, A., Shapiro, A., and Homen-de Mello, T., "The sample average approximation method for stochastic discrete optimization," *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 479–502, 2001.

[48] Kushner, H. and Yin, G., *Stochastic Approximation Algorithms and Applications*. McGraw-Hill, New York, NY, 2004.

[49] Lee, L. and etc., "Approximate simulation budget allocation for selecting the best design in the presence of stochastic constraints," *IEEE Transactions on Automatic Control*, vol. 57, no. 11, pp. 2940–2945, 2012.

[50] Li, J., Sava, A., and Xie, X., "Simulation-based discrete optimization of stochastic discrete event systems subject to non closed-form constraints," *IEEE Transactions on Automatic Control*, vol. 54, no. 12, pp. 2900–2904, 2009.

[51] Nemirovski, A., Juditsky, A., Lan, G., and Shapiro, A., "Robust stochastic approximation approach to stochastic programming," *SIAM Journal on Optimization*, vol. 19, pp. 1574–1609, 2009.

[52] Norkin, V., Ermoliev, Y., and Ruszczyński, A., "On optimal allocation of indivisibles under uncertainty," *Operations Research*, vol. 46, pp. 381–395, 1998.

[53] Norkin, V., Pflug, G. C., and Ruszczyński, A., "A branch and bound method for stochastic global optimization," *Mathematical Programming*, vol. 83, pp. 425–450, 1998.

[54] Pagnoncelli, B., Ahmed, S., and Shapiro, A., "Sample average approximation method for chance constrained programming: Theory and applications," *SIAM Journal on Optimization*, vol. 142, no. 2, pp. 399–416, 2009.

[55] Park, C. and Kim, S., "Handing stochastic constraints in discrete optimization via simulation," *Proceedings of the 2011 Winter Simulation Conference*, pp. 4217–4226, 2011.

[56] Pichitlamken, J. and Nelson, B., "A framework for locally convergent random-search algorithms for discrete optimization via simulation," *ACM Transactions on Modeling and Computer Simulation*, vol. 13, no. 2, pp. 155–179, 2003.

[57] Polyak, B. and Juditsky, A., "Acceleration of stochastic approximation by averaging," *SIAM Journal on Control and Optimization*, vol. 30, pp. 838–855, 1992.

[58] Pujowidianto, N. A., Lee, L. H., Chen, C.-H., and Yap, C. M., "Optimal computing budget allocation for constrained optimization," *Proceedings of the 2009 Winter Simulation Conference*, pp. 584–589, 2009.

[59] Robbins, H. and Monro, S., "A stochastic approximation method," *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400–407, 1951.

[60] Robinson, S. M., "Analysis of sample-path optimization," *Mathematics of Operations Research*, vol. 21, no. 3, pp. 513–528, 1996.

[61] Rubinstein, R. and Kroese, D., *The Cross-Entropy Method*. Springer, New York, NY, 2004.

[62] Shapiro, A. and Wardi, Y., "Convergence analysis of stochastic algorithms," *Mathematics of Operations Research*, vol. 21, no. 3, pp. 615–628, 1996.

[63] Shi, L. and Ólafsson, S., "Nested partitions method for global optimization," *Operations Research*, vol. 48, no. 3, pp. 390–407, 2000.

[64] Shi, L. and Ólafsson, S., "Nested partitions method for stochastic optimization," *Methodology and Computing in Applied Probability*, vol. 2, no. 3, pp. 271–291, 2000.

[65] Sun, L., Hong, L., and Hu, Z., "Balancing exploitation and exploration in discrete optimization via simulation through a Gaussian process-based search," *Operations Research*, vol. 62, pp. 1416–1438, 2014.

[66] Swisher, J., Hyden, P., Jacobson, S., and Schruben, L., "A survey of recent advances in discrete input parameter discrete-event simulation optimization," *IIE Transactions*, vol. 36, pp. 591–600, 2004.

[67] Swisher, J., Jacobson, S., and E., Y., "Discrete-event simulation optimization using ranking, selection, and multiple comparison procedures: A survey," *ACM Transactions on Modeling and Computer Simulation*, vol. 13, pp. 134–154, April 2003.

[68] Xu, J., "Efficient discrete optimization via simulation using stochastic kriging," *Proceedings of the 2012 Winter Simulation Conference*, pp. 466–477, 2012.

[69] Xu, J., Nelson, B., and Hong, L., "An adaptive hyperbox algorithm for high-dimensional discrete optimization via simulation problems," *INFORMS Journal on Computing*, vol. 25, pp. 133–146, Winter 2013.

[70] Yakowitz, A., "A globally convergent stochastic approximation," *SIAM Journal on Control and Optimization*, vol. 31, no. 1, pp. 30–40, 1993.

[71] Yakowitz, A. and Lugosi, E., "Random search in the presence of noise, with application to machine learning," *SIAM Journal on Scientific and Statistical Computing*, vol. 11, no. 4, pp. 702–712, 1990.

[72] Yan, D. and Mukai, H., "Stochastic discrete optimization," *SIAM Journal on Control and Optimization*, vol. 30, pp. 594–612, June 1992.

VITA

Liujia Hu was born in Jingzhou, Hubei Province, People's Republic of China, in 1986. He received his B.A. in Economics and B.S. in Mathematics from Wuhan University in China in 2008, and M.S. in Operations Research from Columbia University in the City of New York in USA in 2010. His research interests are in simulation-based optimization, Monte Carlo simulation, and meta-modeling. After finishing his Ph.D. in Operations Research at Georgia Institute of Technology, he will join Ernst & Young as a Senior Consultant in their Financial Services Office.